Traffic Anomaly Detection and DDOS Attack Recognition Using Diffusion Map Technologies

Michael Zheludev
Qrator Labs
e-mail: qukengue@andex.ru;
mz@qrator.net

Evgeny Nagradov,
Qrator Labs
email: en@qrator.net

ABSTRACT
This paper provides a method of mathematical representation of the traffic flow of network states. Anomalous behavior in this model is represented as a point, not grouped in clusters allocated by the "alpha-stream" process

Keywords
Kernel Methods, Data Analysis, Diffusion Maps

1. INTRODUCTION

Network attacks becoming a major threat on nations, governmental institutions, critical infrastructures and business organisations. Some attacks are focused on exploiting software vulnerabilities to implement denial of service attacks, damage or steal important data, other use a large number of infected machines to implement denial-of-service attacks. In this paper we are focusing on detecting network attacks by detecting the anomalies in network traffic flow data and anomalous behaviour of the network applications. The goal is to detect the beginning of the attack in a real-time and to detect when the system is returned back to the normal state. In this paper we are not focusing on the problem of identifying the source of the attack and the attack mitigation.

The network traffic flow data can be represented by a set of network-level metrics (amount of packets for different protocols, inbound and outbound traffic, etc.) and application-level metrics (like the response duration histogram for web server). These metrics are collected by the traffic analyser at fixed rate. The goal for the state analyser is to detect anomalous network and/or application behaviour basing on these metrics.

The input data for the analyser is statistics matrix that contains a single row for every traffic time slice. Each row contains the network-level and application-level features that come from different scales. This matrix is the input for the intrusion detection processes (both training and detection steps).

Our method has two sequential steps. Study and analysis of the behaviour of networking datasets and projection of data onto a lower dimensional space - training step. This is done once and updated as the behaviour of the training set changes. During this step we can handle corrupted training sets.

The output from the training step enables online detection of anomalies to which we apply automatic tools that enable real-time detection of problems. Each newly arrived datapoint is classified as normal or abnormal.

Analysis of the indicators of network traffic reveals represent normal behaviour as statistically dependent set, grouped in clusters after reduced dimensionality operation, against which the representation of anomalies. Anomalies is not statistical connected with the basic set of states. They appear as distant from the main cluster points.

2. THE TRAFFIC ANALYSER

The traffic analyser processes the network packets and summarises the network-level statistics. These metrics include: tcp flags usage; number of control tcp packets (packets without payload); number of data tcp packets (packets with payload); number of source (client) packets; number of source control packets; number of source data packets; number of source data bytes; number of destination (server) packets; number of destination control packets; number of destination data packets; number of destination data bytes.

TCP-connections could be reassembled to estimate application-level metrics. Another sources of application-level metrics are the log files from applications (like access-logs on HTTP web server). The analyser processes the application logs to collect and summarise application level metrics (like total amount or requests, total amount of errors, histogram of the response times, histogram of error codes, etc). These metrics can be extended by adding other sources of behaviour metrics, like e-mail server logs, database server logs, cpu/memory metrics. We measure, receive and sense many parameters (features) at every pre-determined time interval – forming high dimensional data. The challenge are: How to cluster and segment high-dimensional data? How to find distances in high-dimensional data? How to find deviations from normal behaviour?

Challenge: How to process an “ocean” of data in order to find abnormal patterns in the data? How to fuse data from different sources (sensors) to find correlations and anomalies? How to find distances in high-dimensional data? They do not exist. How can we determine whether a point belongs to a cluster/segment or not? The goal is to identify points that deviate from normal behaviour which reside in the cluster/segment. How we treat huge high dimensional data that is dynamically and constantly changes? How can we model the high dimensional data to find deviations from normal behaviour?

3. DETAILS
The traffic state at each time point can be represented by a vector, as shown in Figure1

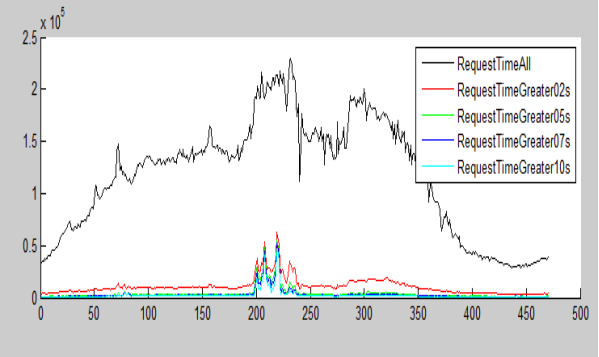


Figure1: traffic behavior in a single day, represented by several factors.

Thus, the traffic can be modeled as a random process related to the vector X (t), where t is time. Define $X=\{X_t\}_t$ the dataset of all traffic states X (t), where for each t X (t) belongs to n-dimensional space R^n. The first goal is: construct such mapping $f: X \subseteq R^n \longrightarrow R^k, k << n$ Such that vectors with regular behavior will be displayed in the compact cluster. The Diffusion Maps (DM) [1] is the manifold learning scheme. DM embeds high dimensional data into an Euclidean space of substantially smaller dimension while preserving the geometry of the data set. The global geometry is preserved by maintaining the local neighborhood geometry at each point in the data set. DM uses a random walk distance that is more robust to noise since it takes into account all the paths between a pair of points. Furthermore, DM can provide parameterization of the data when only a point-wise similarity matrix is available. This may occur either when there is no access to the original data or when the original data consists of abstract objects.

The general idea of constructing DM is the following:

a. Building a graph G on X with a weight function w that corresponds to the local point-wise similarity between the points in X.

b. Construction of a random walk on the graph G via a Markov transition matrix P which derived from W.

c. Spectral decomposition of P.

By designing a local geometry that reflects the needed quantities, it is possible to construct a diffusion operator with spectral decomposition that enables the embedding of X into a space of substantially lower dimension. The Euclidean distance between a pair of points in this space is equal to the random walk distance between the corresponding pair of points in the original space. We shall use the DM for dimensionality reduction and data clustering.

For that purpose the mapping $f: X \subseteq R^n \to R^k, s \le n$ is constructed in [1]. Each vector x is represented via the relationship with other vector from the dataset X. If $X = \{x_1, \dots, x_s\}$, then $f(x_i) = (\rho(x_i, x_1), \dots, \rho(x_i, x_s))$ and $\rho(x_i, x_j)$ measures the relationship between x_i and x_j. The diffusion distance was introduced in [1] and is equal to

$$\rho(x_i, x_j) = 1 - \exp\left(-\frac{\|x_i - x_j\|}{\sigma^2}\right).$$

Closed data-points satisfy $\rho(x_i, x_j) \approx 0$ while the remote data points satisfy $\rho(x_i, x_j) \approx 1$.

Build the Markov matrix P, by normalizing $\rho(x_i, x_j)$

$$p_{ij} = \frac{\rho(x_i, x_j)}{d(x_i)}$$

and

$$d(x_i) = \sum_{x \in X} \rho(x_i, x)$$

Then we find eigenvectors $\psi_j(\mathbf{f})$ and eigenvalues of the λ_j equation

$$P\Psi_j(f) = \lambda_j \Psi_j(f)$$

The (i,j) element of P^t gives the probability of going from node i to node j in t steps.

Apply embedding in the low-dimensional space (diffusion map)

$$\Psi: X \subseteq R^n \longrightarrow R^k, k << n$$

$$\Psi: \mathbf{f} \to (\lambda_1\psi_1(\mathbf{f}), \lambda_2\psi_2(\mathbf{f}), \dots, \lambda_m\psi_m(\mathbf{f}))$$

Spectral fall-off contribute to dimensionality reduction At t>1

$$\Psi_t: \mathbf{x} \to (\lambda_1^t\psi_1(\mathbf{f}), \lambda_2^t\psi_2(\mathbf{f}), \dots, \lambda_m^t\psi_m(\mathbf{f}))$$

For large t, large–scale structures in data can be captured in fewer diffusion coordinates.

According [1], the diffusion distance (the probability of transition from one vertex of the Markov chain G to other) is equal to the distance in the Euclidean metric after diffusion mapping

$$Diff_t(x, y) \approx \|\Psi_t(x) - \Psi_t(y)\|_{R^m}$$

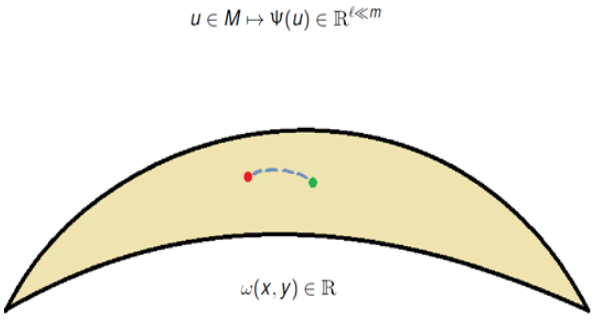


Figure 2. It is easy to see that the map has the following properties:

• The map represents the data in a space of dimension k.

• The map is not linear.

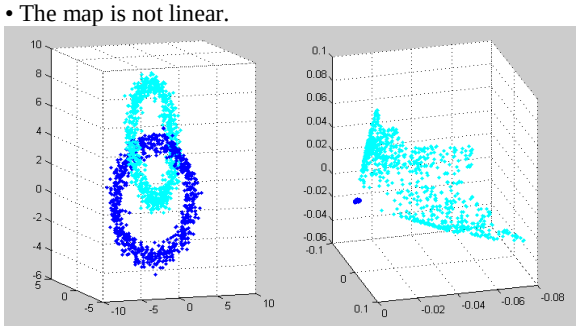


Figure 3. The figure illustrates the effectiveness of the separation of mixed known clusters via "diffusion maps". If the generated data is represented as two interlocking rings (marked different shades of blue), no any linear methods is able to divide it. Nevertheless, a random walk on the graph represented by these rings, have ability to divide the classes. The probability remain inside the same ring by random walk is greater than the probability of jumping from one ring to another. The distance between the images of points is equal

to the diffuse distance, that is, the probability to get from point x to point y via random walk on the graph for the time t.
For diffusion representation of traffic behavior during the day, we use the next kernel function for calculate weight of the relationship between datapoints

$$\rho_{ij} = e^{\frac{\left\|(i-j) \bmod D\right\|^2}{\sigma_1^2}} e^{\frac{\left\|x_i - x_j\right\|^2}{\sigma 2^2}}$$

Where D is the length (in seconds) of the day. So, the diffusion geometry is oriented around a smooth parametric curve as shown in Figure4.

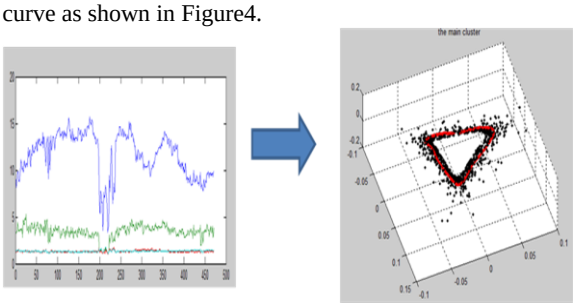


Figure4. The curve represents one day of traffic activities.

After the diffusion map is constructed for the vectors from the training database, it should be extended to arbitrary vector.

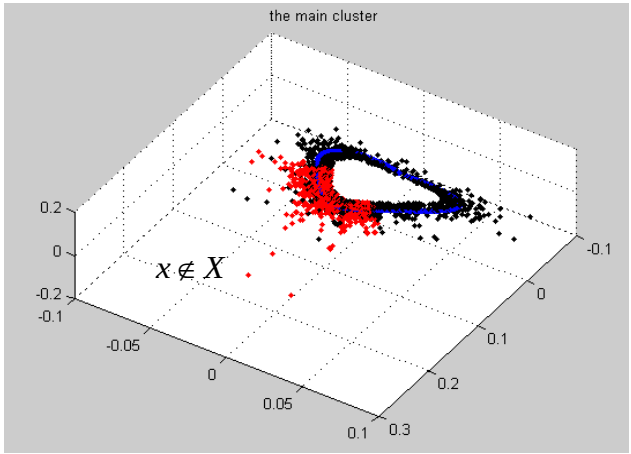
Our Challenge: Once X is mapped - extension of $\bar{f}$ to $x \notin X$, using representatives from X (sampling).
In other words, based on f and the distances of x from X, extend f (denoted by $\bar{f}$) for any $x \notin X$.

Let $x \notin X$. Construct the relationship vector

$$\Xi = (\rho(x, y) \mid y \in X);$$

The extension of diffusion map to x is:

$$\overline{f_t}(x) = \sum_{i=1}^{m} (\Xi * \Psi_i) / \lambda^t{}_i;$$



Once X is mapped - extension of $\overline{f}$ to $x \notin X$, using representatives from X (sampling).
Let E(j) be the approximating curve for manifold in diffusion map represented X and $x \notin X$.
Define homotopy G(x) by nest expression:

$$G(x) = x - \frac{\sum_i E(i)\rho(x, E(i))}{\sum_i \rho(x, E(i))} \qquad (1.1)$$

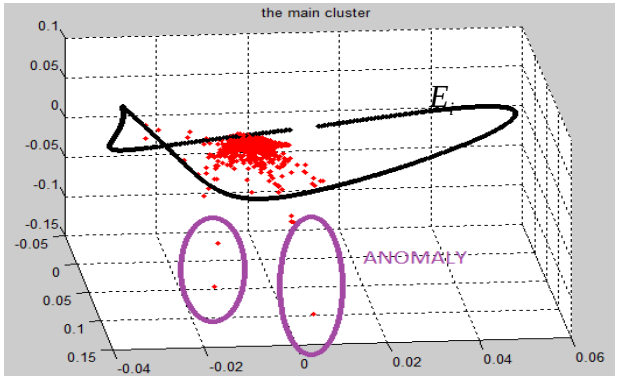


Figure 5. Application homotopy for testing data $x \notin X$ (red color)

This algorithm represents the data so that the vectors with regular behavior are grouped into a Gaussian cluster, whereas abnormal points are located at a considerable distance from the Gaussian.

After this simple algorithm "Alpha-stream" easily determine abnormal points.

Description of "Alpha-stream" algorithm.

The aim is to build an algorithm capable for classifcation of background and anomalies.
Let $X=\{x_1,\ldots x_N\}$ be dataset after "homotopy processing" (1.1) described above. Define measure relationships between datapoints

$$W_{ij} = \exp(-d_{ij}{}^2 / \sigma^2)$$

Where d_{ij} be distance between x_i and x_j and σ be a parameter of the algorithm. Define diagonal matrix D with elements

$$\eta_i = \sum_j W_{ij}$$

The "Laplassian" matrix is the L = D-W. The segmentation problem is to find the function

$$\alpha : X \longrightarrow [0,1],$$

such that if $\alpha(x_i) = 1$ then the datapoint x_i belongs to background. Otherwise, if $\alpha(x_i) = 0$ then the datapoint x_i belongs to foreground. The distribution of α measures the probability of being an anomaly. We will look for the function α by minimizing the energy functional.
The form of the energy functional is as follow

$$F(\alpha) = \sum_i \chi_i^{\Theta} \alpha_i + \sum_{i,j} W_{ij} (\alpha_i - \alpha_j)^2$$

Where χ^{Θ} be the characteristic function of the set, defined as a rough estimate of the background:

$$\Theta = \{x \in X \mid mahal(x, X) < 1\}$$

The first member of the functional declines to take the alpha value of 1 on the set Θ. The second term of the functional calls alpha to change in places of small relationship between the elements W_{ij}.
We can write energy functional in other form

$$F(\alpha) = \chi^{\Theta}\alpha^{T} + \alpha^{T} L\alpha$$

We use gradient descent method to maximize the functional.

$$\frac{\partial F(\alpha)}{\partial \alpha} = \chi^{\Theta} + L\alpha$$

And we have the next iterative process to get distribution of α:

$$\alpha_{i+1} = \alpha_i - \chi^{\Theta} - L\alpha_i$$

Where the initial approximation step for α is random uniform distribution.

4. TEST

During research 4 domains from database were tested. Anomalies were successfully detected by 95%. Example anomaly detection is shown in Figure 6.

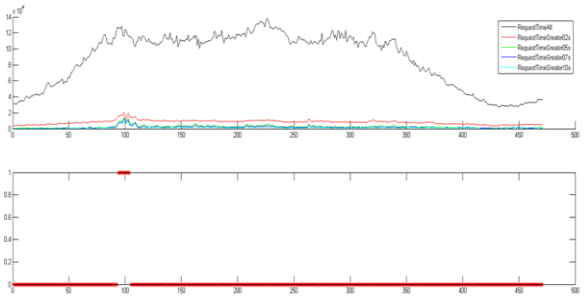


Figure6. Up: traffic behavior in a single day, represented by several factors. Down: red points represent the anomaly activities on traffic.

Comparison of the obtained present method with the projection on the PCA we afford in the form of confusion matrix

Column1	anomalies	background
anomalies	0,95	0,05
background	0,03	0,97

Table 1: distribution of the "false-positive" and "true-negative" for the result of presented algorithm.

Column1	anomalies	background
anomalies	0,63	0,37
background	0,29	0,71

Table 2: distribution of the "false-positive" and "true-negative" for the result of projection on PCA.

REFERENCES

[1] R.R. Coifman, S. Lafon, Diffusion maps, Applied and Computational Harmonic Analysis, 21, 5-30, 2006.

[2] Amir Averbuch* and Michael Zheludev Two Linear Unmixing Algorithms to Recognize Targets Using Supervised Classification and Orthogonal Rotation in Airborne Hyperspectral Images Remote Sens. 2012, 4(2), 532-560; doi:10.3390/rs4020532

[3] Unmixing and Target Recognition in Airborne Hyper-Spectral Images Amir Averbuch , Michael Zheludev & Valery Zheludev Earth Science Research; Vol. 1, No. 2; 2012 ISSN 1927-0542 E-ISSN 1927-0550 Published by Canadian Center of Science and Education