Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 57(71), Fascicola 2, 2012

On Performance of Simple Detection of Pulse-Shaped Anomalies in Data Series from NEAR Network Data Collection Tool

Florin Vancea[1]

Abstract – One class of network intrusion detection systems collects information about traffic and looks for anomalies in the collected data. This paper presents an evaluation of two existing detection methods applied to network data collected by a new tool. The relative performance of each method is discussed in correlation with the specific features of the data series resulting from specialized network traffic collection.

Keywords: IDS, network traffic, anomaly detection, Hurst parameter, false positives

I. INTRODUCTION

Network intrusion detection systems (for short IDS) can be classified in two broad categories: systems that search for known attack patterns inside traffic units (frames, packets, segments) [[1] and systems that attempt to detect abnormal behavior using some metric for the traffic (for example packet count per second) processed with signal processing techniques [2]. The latter presents several specific challenges but promises to detect even anomalies that have not been seen before.

Our focus is on the second class and we have developed a system which attempts to detect the anomalies using signal processing methods. A brief description of the system follows in order to establish the context. This article is however only discussing a particular issue related to detection of pulses.

The traffic should be first captured at properly chosen points in the network (routers, switches, gateways, significant servers and significant user stations). The traffic must be analyzed and significant features extracted. The extraction process must maximize the chance to detect different types of abnormal behavior.

Once relevant features are extracted they are presented as time series to algorithms that each attempt to detect specific types of anomalies. The final step is to correlate the results from series obtained either from different or same collection point and to generate alarms when abnormal situations are detected.

We have developed a tool which captures network traffic and collects information about the traffic as series of values (NEAR – Network Extraction of Anomaly Records) [3]. The agent tool is running on network nodes and uses packet capture on all packets on a given network interface. It analyzes the main packet characteristics and updates counters.

The resulting series of values are analyzed by a centralized facility. The whole system (collecting agents and central analyzer) is based on the following set of anomaly detection principles:

1. Signal to noise ratio principle: In order to detect better the anomaly the traffic feature should be chosen such that the signal to noise ratio should be maximized (anomaly is the signal and normal traffic is noise)
2. Darknet principle: Not all segments of network addressing space are experiencing traffic. Addressing space is taken here in the widest sense. Properly applying cuts and considering directional traffic may improve detection rates.
3. Correlation principle: One event in the network is reflected in several features (eventually collected in several places). A particular event will induce a particular correlation signature.

The agents are extracting traffic features using the above principles and produce series of values. Because it would be impossible to collect counters for all possible types of traffic, NEAR is performing real-time traffic classification and generates at most 64 traces of values spaced at 10 second interval. Some of those series represent globally the whole traffic but most of them reflect only a segment of the entire extended addressing space or traffic purpose, as dictated by the above principles. These series are the subject of the following discussion.

II. TRAFFIC FEATURES

Most of the traffic features extracted are based on packet counters, even if other sources of information can be considered (for example local processor or memory utilization or number of network sockets in

[1] Faculty of Electrical Engineering and Information Technology, Computer and Information Technology Dept. Str. Universităţii 1, Oradea, Romania, e-mail fvancea@uoradea.ro

BUPT

use). Many of the packet-count series are counting by design only a portion of the traffic (for example file-sharing or mail-related or HTTP-related). There are also other series with global significance (for example ARP or ICMP traffic or SYN-only counts).

We have found that the distribution of the resulting series is very diverse with respect to the Hurst parameter.

The current trend in the literature is to consider that network traffic has a self-similar distribution and the Hurst parameter estimated for the distribution is in range 0.7..0.9 [4][5]. However the Hurst parameter is merely estimated, not calculated, and there may be large errors involved [6].

We have used as estimator the function wfbmesti() provided by Matlab which applies both a wavelet-based and a loglog method in order to obtain more than one estimate for the series generated by NEAR..

We took samples of significant series extracted with NEAR from DARPA-98 dataset and from live captures on a real network. Only significant sets were taken into consideration (more than 1000 relevant samples per day, out of a maximum of 8640).

The estimated Hurst parameter was found to be as low as 0.05, with many values grouped below 0.5, which is quite contrary to the classical results above. Our explanation for this is that the distribution of series extracted by NEAR is not aligned with typical global traffic distribution discussed usually in the literature. Some of the possible reasons are (different reasons may apply to different series):

- the series extracted by NEAR are representing only a particular slice of traffic and this is destroying the self-similarity
- the population of participating nodes in our examples for some series was not large enough
- some series are typically produced by machine-to-machine traffic

We have encountered works that support this idea [5][7][8]. The original high Hurst parameter values were found in global traffic counters for aggregated traffic, measured at the boundary of large domains. Many of the further confirmations of that finding were done on aggregate traffic because that is the most readily available type of traffic.

However, when special types of traffic were considered, the self-similarity was not that strong or was not even present at all.

This does not mean that NEAR would not meet in its daily usage series with large values of the Hurst parameter but rather that one should be ready to deal with all types of distributions. Furthermore, it may be better to apply one technique rather than another, based on the grade od self-similarity expected on a particular trace.

This is the reason why the simulations below were done for the behavior of the detection techniques using values of Hurst parameter over the entire 0..1 range.

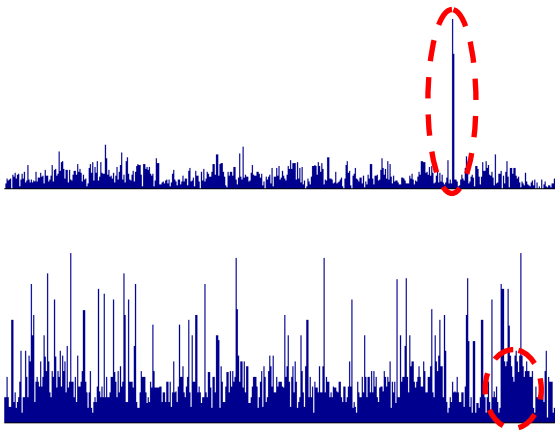


Fig. 1. Examples of pulse anomalies

III. SIMPLE ANOMALY DETECTION

During our research carried out visually over NEAR traces we have identified several types of possible anomalies [3]:

- single pulse present over regular series values. This may be a narrow and rather large spike or a wide platform with smaller amplitude
- periodic perturbation or lack of normal periodicity in the series
- significant values of traffic in grey addressing segments
- correlation-based deviations from normal situation, either in the positive directions (presence when not expected) or in the negative direction (absence when expected)

We are focusing in the following pages only on the simplest case, the pulse perturbation superimposed over normal series, as shown in Fig.1.

Because this is the simplest case we have attempted to use two simple detection methods. Both use preprocessing followed by application of a threshold.

The first method preprocesses the series by applying a filter to eliminate unwanted local maxima. The results below are for a simple moving average with a rectangular window.

The second method uses SNEO (smoothed nonlinear energy operator). The original operator (NEO or TEO from Teager Energy Operator) was introduced by Kaiser [9] and is described by the discrete equation (1) where x_n is the series to analyze.

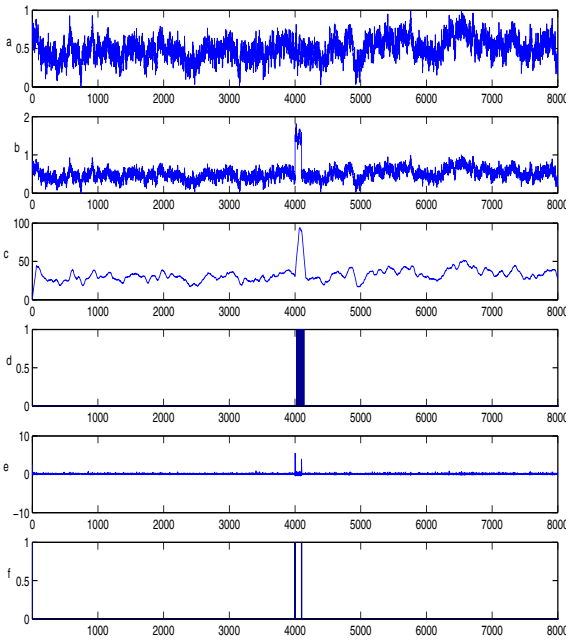
$$y_n = (x_n)^2 - x_{n-1}x_{n+1} \quad (1)$$

The operator enhances the detection process by focusing on the energy of the pulse. Large variations in consecutive values are translated into large values for the y value.

The operator can also be generalized for a k gap (instead of 1) as given by (2)

$$y_{n,k} = (x_n)^2 - x_{n-k}x_{n+k} \quad (2)$$

BUPT



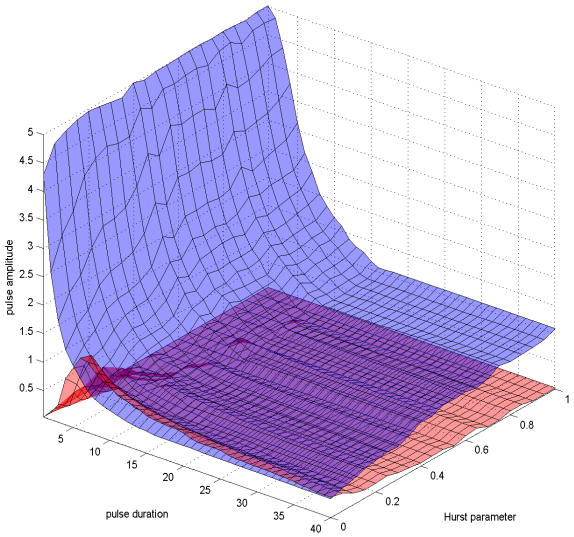


Fig. 3. Filter and threshold for window 16

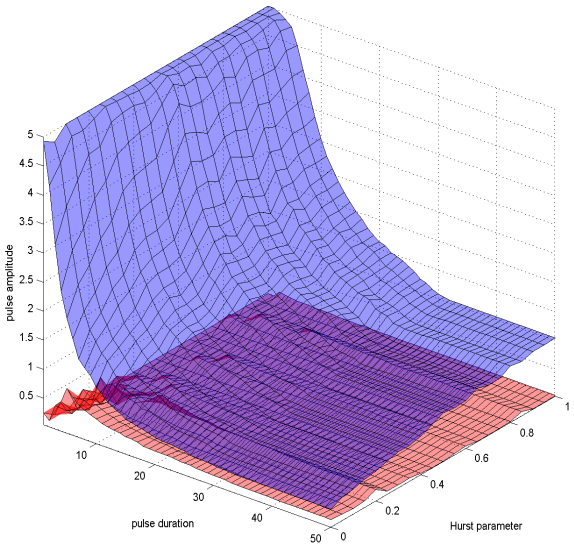


Fig. 4. Filter and threshold for window 32

It is shown [10] that the $y_{n,k}$ is the energy of the zerc phase sinus wave passing through x_{n-k}, x_n, x_{n+k}.
Because the simple operator may introduce fals positives, an additional step is added, smoothing th results by convolution with a smoothing window (i our case, a Bartlett triangular window).
The final threshold in the results below is chosen b an n-sigma over mean value method. The simpl filter-and threshold case uses a 3-sigma level whil the SNEO method can use a 5-sigma level due to th additional sensitivity brought by the special operator.

IV. RESULTS

Both methods were run on synthetic series built a fractional Brownian noise with Hurst factor between 0 and 1. Even if 0 and 1 are limit cases and the performance of the generator is very likely to be questionable at the boundary, the intention was to cover the whole theoretical range of Hurst value.
Since the threshold is mean-dependent, the presence of the pulse impacts the false positive response of the method. Therefore the pulse is placed at the middle of an 8000 samples series, which creates a space to look for false positives before and after the pulse (with a reasonable guard interval). To evaluate the false negative rate, the detector response is expected in a window placed around the original pulse, having the width larger than the pulse in order to account for the lag introduced by filtering and aligned at the left because the detection cannot work “before” the pulse itself.
A sample run is represented in Fig. 2. Trace (a) is the original series and trace (b) is the perturbed series. Trace (c) is the filtered version and trace (d) is the output from the filter and threshold method. Trace (e) is the output from SNEO with Bartlett window and trace (f) is the SNEO detector output. The series amplitudes are normalized and in the sample the impulse amplitude and duration is exaggerated to produce a clearer figure. It is obvious by simple inspection that threshold-and-filter will produce a single wide detection result and the SNEO method will produce at least one sharp peak corresponding to the high energy from the pulse edge.
Our aim is to investigate the response of the two methods for different types of series (different Hurst parameters) and for different impulse widths. In order to do that, the amplitude of the impulse was increased progressively on a quasi-logarithmic scale until the false positives disappeared (due to the threshold being increased) and until correct detection occurred (i.e. there was no false negative). The whole process is bound to a particular realization of the synthetic signal but because we want to determine the statistical values the whole process had to be run several times and the results were averaged.

BUPT

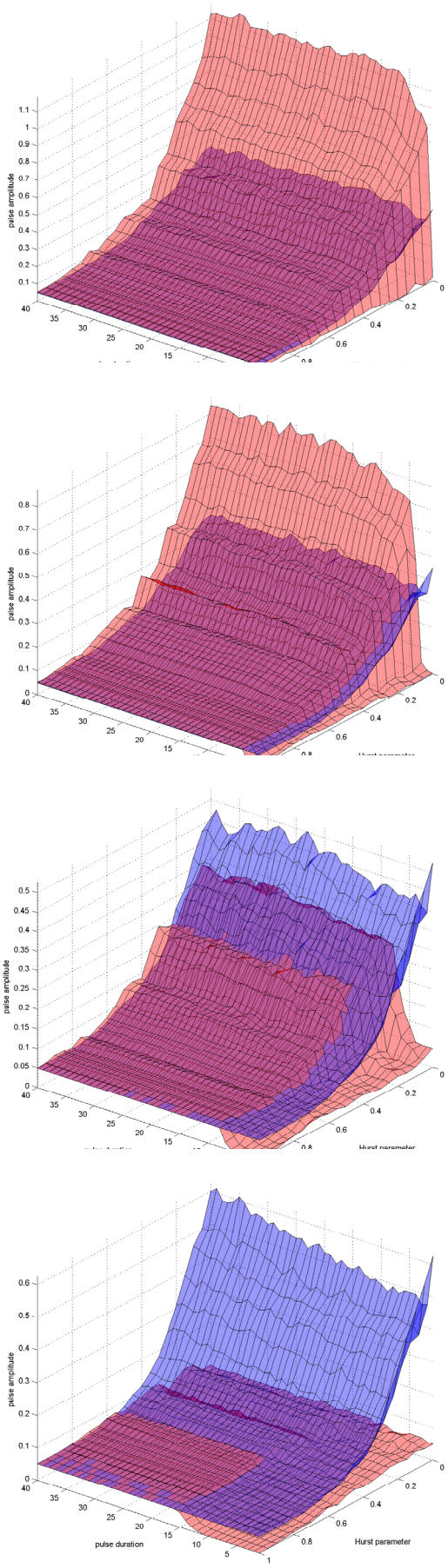


Fig. 5. SNEO results for k=1. From top to bottom the window size is 9, 17, 33, 65

On the available hardware (Intel Core2 T7100 @ 1800 MHz) the above process meant 40 runs to reasonably average particular realizations while keeping the run time below a couple of minutes.

We performed first a couple of runs for the filter and threshold method, represented in Fig. 3 and Fig. 4. The lower surface is the set of values for the false positive threshold and the upper surface is for the false negative threshold (or the detection threshold). One can see that the performance is limited by the detection threshold and that the detection performance degrades strongly for short impulses. Furthermore, after the pulse reaches the width of the averaging window there is no further sensitivity gain. By comparing the two diagrams, it is clear that a narrower window performs poorer than a larger one in terms of false positive rejection at low Hurst values.

After reaching the steady zone, both windows exhibit similar sensitivity, with better behavior for lower Hurst values. The main disadvantage of this method is the poor overall sensitivity and especially poor sensitivity to short pulses (or high rate of false positives for very narrow filters).

The same testing principles were applied when running the SNEO detector. For SNEO it is possible to vary the window width (i.e. change the smoothing factor) or to use different values for k (i.e. change the width tuning factor).

The results of the simulations are presented in Fig. 5 and Fig. 6.

First images, from Fig. 5, present different values for the window width (9, 17, 33 and 65). Last two images (Fig. 6) present the effect of increasing the gap in SNEO to 4 and respectively 16. An important side note is that the axis position for the SNEO set of diagrams is different than the one used in the filter-and-threshold case in order to better show some particularities of these results. The relative position of the surface for false positive thresholds and the surface for detection threshold is also different; we will comment on it for each case.

The first observation is that SNEO sensitivity is significantly better compared to the simple threshold. The maximal values of sensitivity in the filter and threshold case are much larger. The worst sensitivity for SNEO is below 1:1 ratio, actually below 0.4–0.5 in the worst case. Larger window sizes yield worse results and the best detection sensitivity is achieved for large Hurst parameter values. The actual sensitivity may be even better in those cases because first non-zero relative value for the pulse amplitude was 0.05. From the first set of diagrams we can see that the detection sensitivity is similar for a wide range of window values at a given k value.

An interesting result is that there is no clear relation between the thresholds for no false positives and no false negatives. At some points in the window-Hurst plane NFP is the limiting factor, at others NFN is the limiting factor.

The window size is important here. Narrower windows are making the false positive threshold to be

BUPT

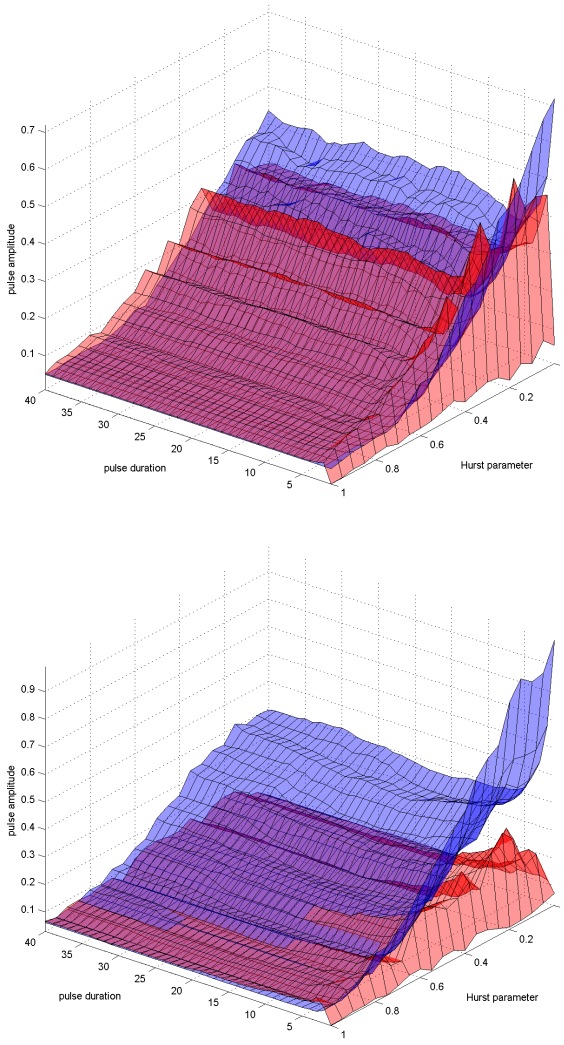


Fig. 6. SNEO results for different *k*. Top is *k*=4, Bartlett-9 and bottom is *k*=16, Bartlett-9

the main limitation factor but as we increase the window size the filtering effect eliminates better the "noise" in series with lower Hurst values. At window size 33 the false positive threshold becomes partially irrelevant and at window size 65 it is completely irrelevant (therefore the detection threshold becomes the main limitation factor).

We can also see the filtering effect of the window in the area corresponding to the narrow impulses (below ¼ of the window size). The false positives are strongly reduced for impulse widths lower than that.

Changing the SNEO gap *k* in its generalized form from equation (2) has the expected results on the detection sensitivity for narrow pulses, specifically for pulses having width shorter than *k*. We can see this as a low-pass filtering effect for components above the natural frequency of the SNEO(*k*) operator. This also brings a positive effect, namely the reduction of the false positive threshold over the entire range. We could not find an explanation for this phenomenon. The simple correlation to the natural frequency of the SNEO(*k*) operator should not be enough because the series are anti-persistent right in the area where there is a significant improvement (low H values).

There is also a local improvement of performance for pulses having width around *k*. The improvement is only marginal due to the simplified application of the operator but the very fact that there is improvement around *k* suggests that some time-scaling technique (or a time-frequency approach) might bring further improvements.

V. CONCLUSIONS

We have studied two simple methods for detecting a pulse anomaly in time series produced by the NEAR intrusion detection system. The novel issue is taking into consideration the Hurst parameter for simulated series, over the whole 0..1 range. This decision was triggered by the fact that we have found estimations of the Hurst parameter in the entire range, for different series (partial traffic versus full traffic series).

As expected, the SNEO operator behaves better than the simple filter and threshold method. A SNEO with reasonable window size is usable for intrusion detection system purposes.

There is also (as expected) some evidence that time-frequency analysis or time-scaling techniques may bring further improvement to pulse detection, at the cost of increased complexity. We will pursue this idea in further work.

The results presented above are applicable to more than simple pulse anomaly detection. The complete system is tracking two more classes of anomalies: periodic anomalies and correlation anomalies. The specific processing techniques applied for those cases will ultimately generate time series where the anomaly is equivalent to a pulse for the duration of the perturbation. This pulse will have to be extracted from the surrounding "noise". An interesting direction of research will be to estimate the self-similarity of the series resulting from those processing techniques, in order to maximize the benefits of the detection methods described above.

REFERENCES

[1] M. Roesch, "Snort - Lightweight Intrusion Detection for Networks", *Proceedings of LISA '99*, p.229-238

[2] P. Barford, J. Kline, D. Plonka and A. Ron, "A Signal Analysis of Network Traffic Anomalies", *PROCEEDINGS OF ACM SIGCOMM INTERNET MEASUREMENTWORKSHOP 2002*,

[3] F. Vancea, C. Vancea, "NEAR - Network Extractor of Anomaly Records or Traffic Split-Counting for Anomaly Detection", *accepted paper to EUROCON 2013, will appear in Conference Proceedings*

[4] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, pp. 1–15, 1994

[5] S. Uhlig and O. Bonaventure, "Understanding the Long-Term Self-Similarity of Internet Traffic", *Proceedings of QOFIS2001*, Coimbra, Portugal, September 2001. Springer-Verlag LNCS2156, pages 286-298

[6] R.G. Clegg, "A PRACTICAL GUIDE TO MEASURING THE HURST PARAMETER", *International Journal of Simulation:*

BUPT

Systems, Science and Technology, ISSN 1473-804x online, 1473-8031 print
[7] S. Katsev, I. L'Heureux, "Are Hurst exponents estimated from short or irregular time series meaningful?", *Computers & Geosciences 29*, Elsevier 2003, 1085–1089
[8] M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes", *IEEE/ACM Transactions on Networking* Vol 5, Number 6, pages 835-846, December 1997
[9] J. F. Kaiser, "On Teager's algorithm and its generalization to continuous signals," *Proc. 4th IEEE Digital Signal ProcessingWorkshop*, Mohonk (New Paltz), NY, Sept. 1990.
[10] I. Obeid, "A WIRELESS MULTICHANNEL NEURAL RECORDING PLATFORM FOR REAL-TIME BRAIN MACHINE INTERFACES", *PhD thesis,*Duke University 2004

BUPT