A Hybrid Approach to Efficient Detection of Distributed Denial-of-Service Attacks

Christos Papadopoulos

Department of Computer Science
Colorado State University
1873 Campus Delivery
USC 228
Fort Collins, CO 80523-1873
E-Mail: christos@cs.colostate.edu

Alexander G. Tartakovsky and Aleksey S. Polunchenko

Center for Applied Mathematical Sciences
Department of Mathematics
University of Southern California
Los Angeles, CA 90089-2532, USA
E-Mail: {tartakov,polunche}@usc.edu

Abstract

An automated system for detecting network traffic anomalies caused by Denial-of-Service attacks is proposed. The system is designed as a two-stage architecture incorporating the change-point detection methodology, used for early attack identification, and further spectral profiling, used for confirmation of the attack presence. The proposed system is shown to be robust and capable of achieving excellent results in terms of first, the speed of detection, and more importantly, the balance between the number of correct detections and the number of false positives. This is accomplished through extensive performance evaluation done using real-world traffic traces containing malicious activity captured at a regional Internet Service Provider (ISP).

1. Introduction

Over the past several years, considerable interest in the field of defense against cyber-terrorism in general, and network intrusion detection in particular, has been induced by a series of external and internal attacks on important corporate and governmental networks, server clusters, and other network resources. For recent reviews, see, e.g., Tartakovsky et al. [1].

Statistically, it is known that the most popular (in terms of the frequency of occurrence) type of attack is that of Denial-of-Serive (DoS) attack [2]. A DoS attack is a malicious attempt by a single person or a group of people to disrupt an online service. A DoS attack could involve a single packet exploiting server software bugs, or a traffic stream with a tremendous number of packets to congest the target’s server or network. The latter is known as a bandwidth attack.
In particular, a Distributed Denial-of-Service (DDoS) attack is a bandwidth attack whose attack traffic is initiated by multiple sources. Such an attack leads to a change in the mean value of the number of packets of a particular type (TCP, ICMP, or UDP) and size. It is therefore intuitively appealing to formulate the problem of detecting attacks as a quickest change-point detection problem: to detect changes in statistical models as rapidly as possible (i.e., with minimal average delays) while maintaining the false alarm rate at a given low level.

However, even though change-point detection methods can be effectively used for the design of anomaly-based Intrusion Detection Systems (IDS), their major drawback is false alarms. If there were a mechanism that could be used as an extra filter to sort out the false alarms outputted by the change-point detection, the performance of the latter would be significantly better. Such a hybrid system is proposed in the present paper.

In this paper, we propose a novel hybrid approach to network intrusion detection aimed at (Distributed) Denial-of-Service attacks. The methodology is based on using the change-point detection methodology for preliminary detection of attacks, and discrete Fourier transform to reveal periodic patterns in network traffic which can be used to confirm the presence of attack. The reason why this will work is that when the victim’s link is congested, TCP packets are sent back-to-back thereby creating a periodicity, which in turn should be reflected in the spectrum. More specifically, schematically the system is shown in Figure 1.

![Figure 1: Hybrid IDS Diagram](image)

The paper is organized as follows. In Section 2, we first present the hybrid system. Section 3 reports the results obtained from an extensive evaluation of the proposed system. Finally, conclusions are drawn in Section 4.

2. The Hybrid Approach

PROVIDE AN OVERVIEW OF RELEVANT WORK

2.1. The Change-Point Approach

In network security applications both pre- and post-change models may be poorly specified. In particular, neither Poisson nor Gaussian models fit well into realistic models for network traffic (see below). When the pre-change and post-change densities are unknown, the (log-) likelihood ratios are also unknown and should be replaced by appropriate score functions \( S_l(n) \) that have negative mean values \( \mathbb{E}_\infty S_l(n) < 0 \) before the change occurs and positive mean values \( \mathbb{E}_k S_l(n) > 0 \) after the change occurs.

While no particular model is being specified for distributions, some assumptions on the change should be made. Indeed, score functions can be chosen in a number of ways, and their selection depends crucially on the type of change that is to be detected. For example, different score functions are used to detect changes in the mean and changes in the variance. In the applications of interest, the detection problem can be usually reduced to detecting changes in mean values (mean shifts).

Let \( \mu_l = \mathbb{E}_\infty X_l(j) \) and \( \theta_l = \mathbb{E}_1 X_l(j) \) denote the pre-change and post-change mean values in the \( l \)-th sensor. Typically, the baseline mean values \( \mu_l \) can be estimated quite accurately in advance while the values...
of \( \theta_l \) are usually unknown and either should be estimated on-line or replaced by reasonable numbers, e.g., by the expected minimal values. In the rest of this subsection we suppose without loss of generality that \( \theta_l > \mu_l \), which is usually the case.

For \( l = 1, \ldots, L \), introduce the following score functions \( S_l(n) = (X_l(n) - \mu_l)/\sigma_l - c_l \), where in the general case \( c_l = c_l(n) \) may depend on past observations, which is desirable to guarantee an adaptive structure of the detection procedure. Here \( \sigma_l \) is the standard deviation. As an option, one may take \( c_l(n) = \epsilon \hat{\theta}_{l,n} \), where \( \epsilon \) is a tuning parameter belonging to the interval \((0,1)\) and \( \hat{\theta}_{l,n} = \hat{\theta}_{l,n}(X_l^n) \) is an estimate of the unknown mean \( \theta_l \). Choosing the latter estimators as well as optimizing the parameter \( \epsilon \) based on the training data are not straightforward tasks, as discussed in detail in [1]. For this reason, it is convenient to set \( c_l(n) = c_l \), where \( c_l \) are positive constants that do not depend on \( n \). Positiveness of \( c_l \) is essential to guarantee the negative value of \( \mathbb{E}_\infty S_l(n) = -c_l \) under the no-change hypothesis. On the other hand, \( c_l \) does not have to be too large in order to guarantee the positive value of \( \mathbb{E}_1 S_l(n) = (\theta_l - \mu_l)/\sigma_l - c_l \) under the alternative hypothesis. A particular choice of \( c_l \) is discussed in [1]. If the above conditions hold, the score-based CUSUM statistic in the \( l \)-th sensor

\[
W_l(n) = \max \{0,W_l(n-1) + S_l(n)\}, \quad n \geq 1, \quad W_l(0) = 0
\]

remains close to zero in normal conditions while when the change occurs it starts rapidly drifting upward.

Likewise, the combined from all the sensors, centralized CUSUM statistic

\[
W(n) = \max \left\{0,W(n-1) + \sum_{l=1}^{L} S_l(n)\right\}, \quad n \geq 1, \quad W(0) = 0
\]

has a similar behavior.

The time of alarm in the centralized detection scheme is defined as the first time \( n \) when the statistic \( W(n) \) crosses a positive threshold.

**2.2. The Spectral Approach**

As a method of confirming that there is indeed an attack taking place Papadopoulos et al. [3] have proposed to consider the spectral characteristics of the traffic. In this section, we outline our methodology for analyzing the spectral characteristics of an attack stream.

The process of obtaining the corresponding spectral representation from the raw packet trace consists of three main steps, which we next explain.

First, we select packet arrivals as the measurable real-world events. The only information we are interested in from the packet trace is the packet arrival times. We extract them from the packet trace to form a sequence of arrival times. In addition, we divide a packet trace into smaller segments of the same length \((l\text{-second long each})\) before extracting packet times from it. The purpose of segmentation is that we can calculate and then compare the spectrum for segments of the same length later. The length of each segment \( l \) is a configurable parameter and we will discuss it further in the next subsection.

Second, we sample each segment with a sampling rate \( p \) to obtain a time series \( X \), where \( X(i) \) is the number of packets that arrive in the time period \([i/p, (i+1)/p)\). The time is relative to the start of the segment, and \( i \) varies from 0 to \( l \times p - 1 \). This results in \( N = l \times p \) number of samples for each segment. In addition, we subtract the mean arrival rate before proceeding with spectral transform in the next step, since the mean value results in a discrete cosine (DC) component in the spectrum that does not provide useful information for our purposes.

For stationary segments, we compute the power spectral density (PSD) by performing the discrete-time Fourier transform on the autocorrelation function (ACF) of the attack stream. The autocorrelation of an attack stream is a measure of how similar the attack is to itself shifted in time by offset \( k \). When \( k = 0 \) we compare the attack stream to itself, and the autocorrelation is maximum and equal to the variance of the attack stream. When \( k > 0 \) we compare the attack stream with a version of itself shifted by lag \( k \). The
autocorrelation sequence $r(k)$ at lag $k$ is

$$c(k) = \frac{1}{N} \sum_{t=0}^{N-k} (X(t) - \bar{X})(X(t+k) - \bar{X}),$$

$r(k) = c(k)/c(0)$

where $\bar{X}$ is the mean of $X(t)$, $N$ is the number of samples and $k$ ranges from $-N$ to $N$. Consequently, the power spectral density $S(f)$ is obtained by applying discrete-time Fourier transform to the autocorrelation sequence of length $M$ and using its magnitude

$$S(f) = \left| \sum_{k=0}^{M} r(k)e^{-2\pi ifk} \right|,$$

where $i = \sqrt{-1}$.

Intuitively, spectrum $S(f)$ captures the power or strength of individual observable frequencies embedded in the time series.

There are two important parameters in the above steps. The first one is the length of each trace slice $l$. If the slice length $l$ is too short, the spectrum will be sensitive to temporary or transient phenomena on the network. If it is too long, the arriving process is unlikely to be stationary.

The other parameter of importance is the sampling rate $p$. Recall that according to the Nyquist-Shannon Theorem, for any given sampling rate $p$, the highest frequency that is observable is $p/2$. In particular, if the sampling rate is too low, aliasing can occur. On the other hand, if it is too high, it will unnecessarily increase both storage and processing overhead. As a rough estimate, for a given link speed and packet size, the maximum required sampling rate can be computed by determining the minimum packet inter-arrival time and sampling at twice that frequency.

3. Experimental Results

This section reports the results obtained from the experimental analysis of the aforementioned hybrid detection system. The dataset used as a test bed is a tcpdump trace file containing a 440-second long fragment of real-world malicious network activity, identified as a DDoS attack. The trace was captured on 06-29-2002 at 15:34:15 UTC off one of the Los Nettos private networks. Specifically, the attack is an ICMP reflector attack (i.e., large number of spoofed IP addresses flooding the victims link with ICMP echo reply packets) that originates within the Los Nettos network starting at 15:35:57 UTC, i.e., about 102 seconds after the beginning of the trace, and lasts for approximately 244 seconds. The victim’s IP address is 89.1.89.241. The number of reflectors is 143.

As the observations – algorithm’s input, $X^i_n$ – we used the cumulative packet rate (number of packets per time unit). Figure 2(a) presents the cumulative packet rate. Figure 2(b) presents the cumulative bit rate (amount of information transmitted per time unit). It is seen that there is a considerable jump in the packet rate at the time moment the attack begins. This change, however, is not seen in the cumulative bit rate diagram.

We used the cumulative packet rate per sampling period, which was chosen 1 ms. Table 1 presents a summary of the packet rate before and after the attack.

Figure 3 illustrates the pre- and post-attack (change) distributions of packet rate. One may suggest that there is a certain level of resemblance with the Gaussian distribution.

However, the corresponding qq-plots shown in Figure 4 disprove the seeming “Gaussianness” by revealing considerable nonlinearities in the relation of the packet rate distribution and the Gaussian distribution (with the proper mean and variance) to one another.

---

1Los Nettos is a regional Internet Service Provider in Los Angeles. See www.ln.net for details.
Figure 2: Network Traffic Flow

<table>
<thead>
<tr>
<th></th>
<th>Pre-Attack</th>
<th>Post-Attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min Packet Rate</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Max Packet Rate</td>
<td>61</td>
<td>69</td>
</tr>
<tr>
<td>Average Packet Rate</td>
<td>13.327</td>
<td>17.568</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>4.618</td>
<td>6.793</td>
</tr>
<tr>
<td>Number of samples</td>
<td>101847</td>
<td>252029</td>
</tr>
</tbody>
</table>

Table 1: Dataset Packet Rate Summary

Figure 3: Cumulative Traffic Packet Rate Distribution

Figure 5 illustrates the operating characteristics, the AD2D versus \(-\log(\text{FAR})\) for different values of \(c\), the design parameter.\(^2\) The optimal performance of the algorithm is characterized by small AD2D and small FAR. The range of \(-\log(\text{FAR})\) is from 0 to 5, which corresponds to the frequency of false alarms approximately one every 7 ms to one every second. The trade-off between AD2D and FAR may be seen through the following observation: In the far left part of Figure 5, AD2D is small while FAR is very large. This means that every packet causes a false alarm. In contrast, in the far right part, FAR is small while

\(^2\)Recall that FAR is defined as \(\text{FAR} = 1/\text{ARL}_{2\text{FA}}\), so that \(-\log(\text{FAR}) = \log(\text{ARL}_{2\text{FA}})\).
AD2D is large.

The performance of the algorithm rides on the parameter $c$. It can be adjusted to get the lowest AD2D for a given FAR. For this particular experiment, the optimal value of $c$ is about 1.5.

Figure 5 shows the relation between threshold values and AD2D for different values of the parameter $c$. It can be seen that AD2D is a linear function of the threshold for practically any value of $c$. Figure 6(b) shows the relation between thresholds and $\log(FAR)$ for different values of $c$. This relationship is also approximately linear for reasonably large thresholds.

As it can be seen, the parameter $c$ is adjusted manually to achieve the best performance possible. By running several CUSUM’s in parallel, each tuned to its own value of $c$, and applying M-CUSUM we can completely take $c$ out of consideration, while keeping the performance characteristics similar to those discussed above. We can recommend implementing the M-CUSUM procedure that uses a set of reference
points $c_i, i = 1, \ldots, M$ in operating IDS-s with an additional possible manual tuning for an optimal value of $c$. This procedure is robust and very efficient.

Experimentally, by applying the proposed nonparametric version of the change-point detection algorithm to a real DDoS attack, we have shown that the algorithm allows for a reliable automated detection of such attacks in their early stages. However, the major problem is false alarms. To overcome the latter issue it has been proposed to use spectral-based methods. Under attack TCP starts sending packets back-to-back thereby causing periodicity or patterns in the traffic behavior, which in turn can be revealed by applying spectral methods.

Figure 7 gives an example of FFT (fast Fourier transform) in action applying to the above dataset. Notice that in the no-attack mode there are no periodic patterns in the traffic distribution, while under the attack there is a contrast peak suggesting that this might indeed be an attack. This phenomenon can be used to filter false positives and develop a hybrid anomaly-signature intrusion detection system with rejection of false alarms and confirmation of true attacks. It is known that separately Anomaly and Signature IDS-s have pros and cons, and combining them in one unit will allow us to obtain the best possible performance.

Figure 8 presents the interface of the hybrid IDS in action. The first plot shows raw data (packet rate). The second plot shows the behavior of the CUSUM statistic, which is being restarted from scratch (repeated)
when a threshold exceedance occurs. The third plot shows PSD (power spectral density) at the output of the spectral analyzer: the peak appears only when the attack starts (which confirms the attack), while previous threshold exceedances (false alarms) are rejected by the spectral analyzer.

Figure 8: Interface of Hybrid IDS in Action.

4. Conclusion

final remarks go here

Acknowledgments

1. provide info as to who we were supported by financially.

2. mention LANDER, the source we obtained the traces to experiment with from.

References

