Proceedings of the 2001 IEEE
Workshop on Information Assurance and Security
United States Military Academy, West Point, NY, 5–6 June 2001

A novel approach to detection of “denial–of–service” attacks via adaptive sequential and batch–sequential change–point detection methods

Rudolf B. Blažek, Hongjoong Kim, Boris Rozovskii, and Alexander Tartakovsky

Abstract— **In computer networks, large scale attacks in their final stages can readily be identified by observing very abrupt changes in the network traffic, but in the early stage of an attack, these changes are hard to detect and difficult to distinguish from usual traffic fluctuations. In this paper, we develop efficient adaptive sequential and batch-sequential methods for an early detection of attacks from the class of “denial–of–service attacks”. These methods employ statistical analysis of data from multiple layers of the network protocol for detection of very subtle traffic changes, which are typical for these kinds of attacks. Both the sequential and batch-sequential algorithms utilize thresholding of test statistics to achieve a fixed rate of false alarms. The algorithms are developed on the basis of the change-point detection theory: to detect a change in statistical models as soon as possible, controlling the rate of false alarms. There are three attractive features of the approach. First, both methods are self-learning, which enables them to adapt to various network loads and usage patterns. Second, they allow for detecting attacks with small average delay for a given false alarm rate. Third, they are computationally simple, and hence, can be implemented on line. Theoretical frameworks for both kinds of detection procedures, as well as results of simulations, are presented.**

Keywords— **Attack Detection, Change Point Detection, Denial of Service, Network Security, Network Traffic, Service Survivability.**

I. Introduction

In computer networks, large scale attacks in their final stages can readily be identified by observing very abrupt changes in the network traffic. But in the early stage of an attack, these changes are hard to detect and difficult to distinguish from usual traffic fluctuations. In this paper, we develop efficient adaptive batch and sequential type methods for an early detection of attacks from the class of “denial–of–service attacks” (DoS).

Existing intrusion detection systems can be classified as either Signature Detection Systems or Anomaly Detection Systems. Signature Detection Systems detect attacks by comparing the observed patterns of the network traffic with known attack templates (signatures) from a database. If the true attack belongs to the class of attacks listed in the database, then it can be successfully detected and moreover identified. Anomaly Detection Systems compare the parameters of the observed traffic with ‘normal’ network traffic. The attack is declared once a deviation from a normal traffic is observed. Both classes of systems have pros and cons. A detailed discussion can be found in [7].

Our approach belongs to the latter class, i.e. to the Anomaly Detection Systems. The idea of the approach is based on the observation that an attack leads to relatively abrupt changes in statistical models of traffic compared to the “normal mode”. This changes occur at unknown points in time and should be detected ‘as soon as possible’. Therefore, the problem of detecting an attack can be formulated and solved as a change-point detection problem: detect a change in the distribution (model) with a fixed delay (batch approach) or minimal average delay (sequential approach), controlling the rate of false detections (the false alarm rate). See [5],[8]-[20] for relevant results of change-point detection theory. In addition, we combine both methods (batch and sequential) in one unit to develop a multistage (batch–sequential) detection algorithm which turns out to be more robust and reliable to some expense of the detection delay.

The proposed detection methods employ statistical analysis of data from multiple layers of the network protocol for detection of very subtle traffic changes, which are typical for these kinds of attacks. Both the batch and sequential methods utilize thresholding of test statistics to achieve a fixed rate of false alarms. In addition, these methods can be used for detection of generalized denial–of–service attacks consisting of combinations thereof.

The developed detection algorithms have several attractive features. First, they have manageable computational complexity, and hence, can be implemented on-line. Second, both algorithms are self-learning, which enables them to adapt to various network loads and usage patterns. In addition, the sequential algorithm has optimal properties among the totality of algorithms with a pre-specified false alarm rate. It is the quickest detection algorithm in the sense that it minimizes the average delay of detection of an attack for a given false alarm rate.

In addition, when augmented with the feedback about false alarms from an appropriate decision–making authority monitoring the quality of the provided service, these adaptive methods can be used to predict traffic overflows and resource hogging. This will allow the system to dynam-

All authors are affiliated with the Center for Mathematical Sciences, University of Southern California, Los Angeles, U.S.A. E-mail: blazek@math.usc.edu

ISBN 0-7803-9814-9/$10.00 ©2001 IEEE

ically manage the available resources to ensure the survivability of the service.

In contrast to [12] and many other works where parametric models (including hidden Markov models) have been considered, we use a nonparametric approach that has much more robust properties.

II. Overview of Sequential and Batch Detection Methods

The main idea is that the structure of an information system can be described by a stochastic model, and that a failure or an attack leads to an abrupt change of the structure. This change occurs at an unknown point in time. There are two main approaches to detecting such an event, the *fixed–size batch detection* (or *a posteriori* methods), and *sequential change–point detection* [5]. In the latter setting, the problem is formulated as a quickest detection problem: detect a change in the model as soon as possible after its occurrence. This approach is based on the change–point detection theory [5], [14], [16], which is a generalization and modification of Wald's sequential analysis (hypotheses testing).

Both approaches involve two performance indices to measure the performance of the method: the *rate of false alarms* and the *detection delay.* In contrast to the fixed–size *a posteriori* methods, *the sequential detection algorithms minimize the average detection delay for a given false alarm rate.* Therefore, the sequential approach is preferable.

There are two major sequential change-point detection algorithms on the market today: the CUSUM detection procedure introduced by Page [10] and the Shiryaev-Pollak detection procedure [11]-[14]. It is known that both detection algorithms are optimal when the observations are independent and identically distributed (i.i.d.) in pre-change and post-change modes. To be specific, in the conventional setting of a change-point detection problem, one assumes that the observed random variables $X_1, X_2, \dots$ are i.i.d., until a change occurs at an unknown point in time λ, $\lambda \in \{1, 2, \dots\}$. After the change occurs, the observations are again i.i.d. but with another distribution. In other words, it is assumed that $X_1, X_2, \dots$ are independent with $X_n \sim p_0$ for $n < \lambda$ and $X_n \sim p_1$ for $n \geq \lambda$, where $p_0(x)$ and $p_1(x)$ are pre-change and post-change probability density functions, respectively.

Let $\boldsymbol{P}_k$ denote the probability measure that corresponds to the sequence $\{X_n, n \geq 1\}$ when the change occurs at time $\lambda = k$ and let $\boldsymbol{E}_k$ stand for the corresponding expectation. Note that $\boldsymbol{P}_\infty = \boldsymbol{P}_0$, where $\boldsymbol{P}_0$ is a pre-change distribution, and $\boldsymbol{E}_0$ will be used to denote $\boldsymbol{E}_\infty$ (nothing changes). The quickest detection task typically involves the optimization of the trade-off between a measure of the detection delay and a false alarm rate. In mathematical terms, a sequential detection procedure is identified with a stopping time τ for an observed sequence $\{X_n\}_{n \geq 1}$, i.e. τ is an integer-valued random variable, such that the event $\{\tau \leq n\}$ depends on $\boldsymbol{X}^n = (X_1, \dots, X_n)$ and does not depend on X_k for $k \geq n+1$. A "good" detection procedure should have in some sense a low rate of false alarms and small values of $D_\lambda(\tau) = \boldsymbol{E}_\lambda(\tau - \lambda \mid \tau \geq \lambda)$. That is, the average detection time (expected detection lag) should be small provided that there is no false alarm.

For i.i.d. models, the log-likelihood ratio based on the data $\boldsymbol{X}^n$ for testing the hypothesis H_λ that a change occurred at time λ against H_0, that there is no change is equal to

$$Z_{n,\lambda} = \sum_{k=\lambda}^{n} \log \frac{p_1(X_k)}{p_0(X_k)}, \quad n \geq \lambda.$$

Page's (CUSUM) procedure is the stopping time

$$\tau_{\mathrm{CU}} = \tau_{\mathrm{CU}}(h) = \min\{n \geq 1 : U_n \geq h\} \tag{1}$$

where $U_n = \max_{0 \leq \lambda \leq n} Z_{n,\lambda}$ and h is a positive number (threshold). Thus, Page's procedure is motivated by a maximum likelihood argument. Note that the statistic U_n obeys the recursion

$$U_{n+1} = \max\left\{0, U_n + \log \frac{p_1(X_{n+1})}{p_0(X_{n+1})}\right\}, \quad U_0 = 0. \tag{2}$$

In contrast, the Shiryaev-Pollak procedure is motivated by Bayesian, rather than maximum likelihood, considerations. It is identified with the stopping time

$$\tau_{\mathrm{SP}} = \tau_{\mathrm{SP}}(h) = \min\{n \geq 1 : \log R_n \geq h\}, \tag{3}$$

where $R_n = \sum_{\lambda=1}^{n} \exp\{Z_{n,\lambda}\}$. Note that the statistic R_n satisfies the recursion

$$R_{n+1} = (1 + R_n) \times \frac{p_1(X_{n+1})}{p_0(X_{n+1})}, \quad R_0 = 0. \tag{4}$$

Both detection methods (1) and (3) minimize the average detection delay $\sup_\lambda D_\lambda(\tau)$ among the totality of detection algorithms for which the mean time between false alarms is fixed at a given level T, i.e. $\boldsymbol{E}_0\tau = T$ (or, alternatively, the average frequency of false alarms $1/\boldsymbol{E}_0\tau$ is equal to $1/T$). In this case, the threshold values should be chosen from the conditions $\boldsymbol{E}_0\tau_{\mathrm{CU}}(h) = T$ and $\boldsymbol{E}_0\tau_{\mathrm{SP}}(h) = T$. In the preliminary engineering computations one can put $h = \log T$, which guarantees the inequalities $\boldsymbol{E}_0\tau_{\mathrm{CU}} \geq T$ and $\boldsymbol{E}_0\tau_{\mathrm{SP}} \geq T$.

The i.i.d. assumption is very restrictive for many applications, including the ones we are dealing with. Recent advances in the general change-point detection theory [8], [20] allow us to design similar detection procedures which preserve the optimality property for general statistical models (no i.i.d. assumption is involved) when the false alarm rate is low (T is high). The corresponding sequential procedures being optimal (at least asymptotically) at the same

ISBN 0-7803-9814-9/$10.00 ©2001 IEEE

time have manageable computational complexity. The latter features make them very attractive for our applications where the data is very often correlated and almost always non-stationary, even bursty, due to substantial temporal variability.

In contrast to the sequential change-point problem where the ‘homogeneity’ hypothesis is tested on line in the process of data acquisition, the *a posteriori* change-point problem is considered on the fixed-time interval $1, \dots, n_0$. In this case, a good detection procedure is based on the comparison of the statistic $U_{n_0} = \max_{0 \le \lambda \le n_0} Z_{\lambda,n}$ with a threshold h at moment n_0. The decision that a change occurred is made if $U_{n_0} \ge h$. The threshold h is chosen from the condition $\boldsymbol{P}_0(U_{n_0} \ge h) = \alpha$, i.e. such that the false alarm probability is equal to a given value α. If the change occurs at the point λ, then any fixed-size (batch) method detects this change with the fixed delay $n_0 - \lambda$, which is large in all cases where n_0 is large and λ is small. The advantage of the sequential methods is obvious.

In many applications, it can be beneficial to combine both methods by grouping the data obtained in the fixed size intervals and first performing an intra-processing of the data in these fixed-size intervals. Then, the results of this intra-processing are further processed sequentially. A resulting procedure will represent a multistage sequential procedure with batch processing within individual stages. The idea is similar to group sequential tests. The corresponding detection method will be called the *Batch–Sequential Method*.

III. Measurable Characteristics of the Network Traffic Flow

While monitoring network traffic, one can observe various kinds of information related to the headers, sizes, and other characteristics of the received and transmitted packets, as well as the usage of system resources, service quality, and similar aspects associated with the utilization of the network and available resources. For example, in the transport layer, we observe the number of TCP packets categorized by size or type (ACK, SYN, URG etc.), the numbers of UDP packets and their sizes, the source and destination port for each packet, etc. In the application layer it is assumed to observe information about packets associated with a given application. Also of interest is the size of buffers related to received and sent SYN packets and similar information. Among other important characteristics of the network traffic flow are, for example, service delay and percentage of expired requests sent to a monitored network server.

Although the methods described in this paper are general, we will, for the sake of simplicity, discuss their development and application in the realm of our current experiments which are described in more detail below. In our simulations, we *simultaneously* observe the following statistics related to the network flow during a time interval T_k:

$N^{k,i}_{\mathrm{pt}}$ — total number of packets of type pt with sizes in i-th bin received during the k-th time interval, and

B^{k}_{SYN} — SYN packet induced buffer size observed at the end of the k-th time interval,

where the packet type pt is either ICMP, UDP, or TCP. For each packet type, the packets are categorized by their size into a a number of size bins; the time intervals

To be more precise, let $T_1, \dots, T_M$ be fixed size non-overlapping time intervals which form a partition of a fixed time interval T, and let $X^{k,1}_{pt}, \dots, X^{k,n^k_{pt}}_{pt}$ be random variables representing the sizes of n^k_{pt} packets of type pt received by the monitored server during a particular time interval T_k. For the i-th size bin from a fixed partition of $\mathbb{R}^+ = [0, \infty)$ into M_{pt} intervals $A^1_{pt}, A^2_{pt}, \dots, A^{M_{pt}}_{pt}$, we then define the number of packets of type pt with sizes in the bin A^i_{pt} observed during the interval T_k as the corresponding interval frequency

$$N^{k,i}_{pt} = \mathrm{card}\{j : X^{k,j}_{pt} \in A^i_{pt}, j = 1, \dots, n^k_{pt}\},$$

where $i = 1, \dots, M_{pt}$.

IV. The Developed Detection Algorithms

In the following two subsections, we describe the two developed detection algorithms: purely sequential algorithm and batch–sequential algorithm.

A. Sequential Detection Algorithm

If both the pre-change $\boldsymbol{P}_0$ and the post-change $\boldsymbol{P}_1$ distributions are exactly known, then the optimal procedure represents a thresholding of either the CUSUM statistic or the Shiryaev statistic (see (1) – (4)). In our applications, however, it is very difficult, if not impossible, to build an exact model. As a result, the post-change distribution $\boldsymbol{P}_1$ is usually unknown. By this reason, we will use a nonparametric approach.

To be specific, we use simultaneous thresholding of the statistics

$$S^{k,i}_{pt} = \max\{0, S^{k-1,i}_{pt} + N^{k,i}_{pt} - m^{k,i}_{pt} - c_{k,i}\}, \qquad (5)$$

where pt is one of the three packet types (ICMP, UDP, or TCP), i corresponds to the i-th bin A^i_{pt}, $m^{k,i}_{pt}$ represents a historical estimate of $\boldsymbol{E}_0(N^{k,i}_{pt})$, and $c_{i,k}$ is a deterministic sequence that is chosen experimentally to minimize the average detection delay ($c_{k,i} \ge 0$). If any statistic $S^{k,i}_{pt}$ exceeds a threshold h^i_{pt}, then an alarm message is sent to the decision making engine. In other words, the stopping time is defined as

$$\tau = \min_{pt,i} \tau^i_{pt}, \quad \text{where} \quad \tau_{pt} = \min\{k : S^{k,i}_{pt} \ge h^i_{pt}\}. \qquad (6)$$

In the case of the observed buffer sizes B^k_{SYN}, we use a similar statistic $S^k_{\mathrm{SYN}} = \max\{0, S^{k-1}_{\mathrm{SYN}} + B^k_{\mathrm{SYN}} - m^k_{SYN}\}$. Note that the detection algorithm (5), (6) is sensitive to changes in the average intensity of the observed traffic.

To fix the false alarm rate, the thresholds h^i_{pt} are chosen from the conditions $\boldsymbol{E}_0 \tau^i_{pt} = T$, where T is a given number. Note that the value of $1/T$ can be regarded as the frequency of false alarms. An alternative, and even more practical way of stabilizing the false alarm rate is to fix the probability $\boldsymbol{P}_0(S^{n+k,i}_{pt} \geq h^i_{pt}$ for $k = 1, \ldots, n_0) = \alpha$ of a false alarm in the time interval of the length n_0 (for any n). This last constraint is especially important for a batch–sequential algorithm discussed below.

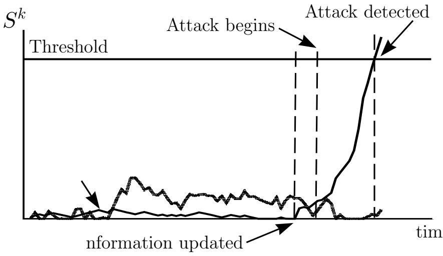

Fig. 1. An illustration of sequential change–point detection. Here, one particular run of a simulated UDP DoS attack is shown. The detection delay is measured in seconds of the simulated time.

As shown in Fig. 1, the information about the patterns of regular traffic flow is updated when a statistic $S^{k,i}_{pt}$ reaches and departs the 0 level. If the decision making engine reports that a previously issued alarm message was a false alarm, then the information about regular traffic patterns and thresholds will be updated accordingly. In other words, in this case the traffic monitoring starts all over again.

B. Batch–Sequential Algorithm

In addition to purely sequential method, we also use a combination of the batch and sequential methods to obtain more robust performance in terms of the false alarm rate.

To explain the batch–sequential method, we will first need to discuss some preliminary aspects of the batch statistic on which it is based. It is well known that if the random variables $X^{k,j}_{pt}$ are independent and identically distributed (i.i.d.) with

$$p^i_{pt} = P(X^{k,1}_{pt} \in A^i_{pt}), i = 1, \ldots, M_{pt}, \tag{7}$$

and if the number of observed packets n^k_{pt} is non–random, then the statistic

$$\chi^2_{pt,k} = \sum_{i=1}^{M_{pt}} \frac{(N^{k,i}_{pt} - n^k_{pt} p^i_{pt})^2}{n^k_{pt} p^i_{pt}} \tag{8}$$

has asymptotically the χ^2 distribution with $M_{pt} - 1$ degrees of freedom, as $n^k_{pt} \to \infty$. This fact is the basis of the traditional goodness–of–fit test of the null hypothesis that, for a fixed M_{pt}-tuple of non-negative numbers $p^1_{pt}, \ldots, p^{M_{pt}}_{pt}$ with $\sum_{i=1}^{n^k_{pt}} p^i_{pt} = 1$, the distribution of $X^{k,1}_{pt}$ satisfies the condition (7) versus the composite alternative hypothesis that (7) does not hold.

When applied to network traffic analysis, this method can be used for detecting anomalies by testing whether the network traffic departed from a known distribution, or from a previously observed empirical distribution. The difficulties arising from the fact that the number of observed packets is random can be overcome when the number of received packets is independent of their sizes. If the observed packet sizes are not independent, the threshold for the hypothesis test may have to be determined using resampling techniques or similar methods.

More generally, if the distribution L^k_{pt} of $X^{k,1}_{pt}$ is known to belong to a parametric family of distributions $\mathcal{L}^k_{pt} = \{L^{k,\boldsymbol{\theta}}_{pt} : \boldsymbol{\theta} \in \Theta\}$, where the probabilities p^i_{pt} in (7) can be expressed as special smooth (see [1], [4]) functions $p^i_{pt}(\boldsymbol{\theta})$ of the true value of the parameter $\boldsymbol{\theta}$, then (8) can be rewritten as

$$\chi^2_{pt,k}(\hat{\boldsymbol{\theta}}) = \sum_{i=1}^{M_{pt}} \frac{(N^{k,i}_{pt} - n^k_{pt} p^i_{pt}(\hat{\boldsymbol{\theta}}))^2}{n^k_{pt} p^i_{pt}(\hat{\boldsymbol{\theta}})} \tag{9}$$

where

$$\hat{\boldsymbol{\theta}} = \operatorname{argmin}\{\chi^2_{pt,k}(\boldsymbol{\theta}) : \boldsymbol{\theta} \in \Theta\} \tag{10}$$

is an estimator of the unknown vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m)'$ with m components. Under the same assumptions of i.i.d. observations and non–random total number of observed packets n^k_{pt}, the statistic $\chi^2(\hat{\boldsymbol{\theta}})$ will have asymptotically χ^2 distribution with $k - m - 1$ degrees of freedom, as $n^k_{pt} \to \infty$. In many situations, the minimizer $\hat{\boldsymbol{\theta}}$ can be found using iterative minimizing procedures. A test can then be constructed for the null hypothesis that the distribution L^k_{pt} of $X^{k,1}_{pt}$ belongs to the family $\mathcal{L}^k_{pt}$ against the alternative $L^k_{pt} \notin \mathcal{L}^k_{pt}$. This method is known to work for a number of families of probability distributions, but in general difficulties arise when the dependency of the probabilities $p^i_{pt}(\boldsymbol{\theta})$ on the parameter $\boldsymbol{\theta}$ violates the smoothness assumptions, or when it cannot be explicitly expressed, making the minimization in (10) hard or impossible.

In terms of network traffic models, such tests would allow for detection of traffic anomalies by testing whether the observed traffic complies to a distribution from a 'smooth' parametric family of network traffic distributions. However, theoretical construction of traffic models corresponding to such smooth parametric families of network traffic distributions is a complex open problem which is beyond the scope of this paper. Another important task is to train the system to create a set of feasible empirical distributions of network traffic under various conditions, and then approximate it with a 'smooth' parametric family which will be used in place of the family $\mathcal{L}$.

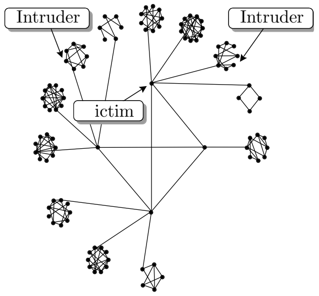

Fig. 2. Transit–stub network topology used in simulations.

Now everything is prepared to describe the batch–sequential algorithm. The algorithm represents the following multi-stage procedure. In the $k-$th stage, we group and process the data observed during the interval T_k and form a batch statistic $\chi^2_{pt,k}$ defined in (8) or, more generally, by (9). For each packet type pt (i.e. ICMP, UDP, or TCP) at stage k, we use for thresholding CUSUM-type statistics $S_{pt,k}$ that obey the recursions

$$S_{pt,k} = \max\{0, S_{pt,k-1} + \chi^2_{pt,k} - \mu_{pt,k}\}, \quad S_{pt,0} = 0,$$

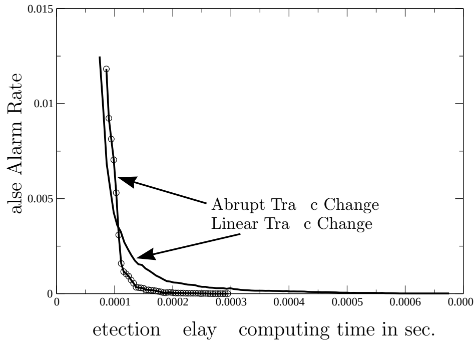

Fig. 3. A summary of the relationship between the detection delay and the false alarm rate for the SYN DoS attack.

where $\mu_{pt,k}$ is a constant whose value is based on the recent history of the network traffic and in general larger than the historical mean of $\chi^2_{pt,k}$. In our preliminary simulations we use, for the sake of simplicity, the statistic $\chi^2_{pt,k}$ as defined by (8) with the sub–optimal choice of $\mu_{pt,k}$ equal the historical mean of $\chi^2_{pt,k}$.

The attack is declared at the time moment $\tau = \min_{pt} \tau_{pt}$, where

$$\tau_{pt} = \min\{k : S_{pt,k} \geq h_{pt}\}.$$

The threshold h_{pt} is chosen from the condition

$$\boldsymbol{P}_0(\tau_{pt} \leq L) \leq \alpha.$$

In other words, the probability of a false alarm $P_{\mathrm{FA}}(L) = \boldsymbol{P}_0(\tau_{pt} \leq L)$ in the interval of the fixed length L is constrained by the given value α. If the distribution of the statistic $\chi^2_{pt,k}$ can be approximated by a χ^2-distribution with M_{pt} degrees of freedom, $\boldsymbol{P}_0(\chi^2_k \leq x) = G_{M_{pt}}$ (see above), then $P_{\mathrm{FA}}(L) \geq 1 - G_{M_{pt}L}(h_{pt})$. The exact value of $P_{\mathrm{FA}}(L)$ is equal to

$$P_{\mathrm{FA}}(L) = 1 - \boldsymbol{P}_0(S_{1,pt} < h_{pt}, \ldots, S_{L,pt} < h_{pt}),$$

and can be computed using the results of renewal theory.

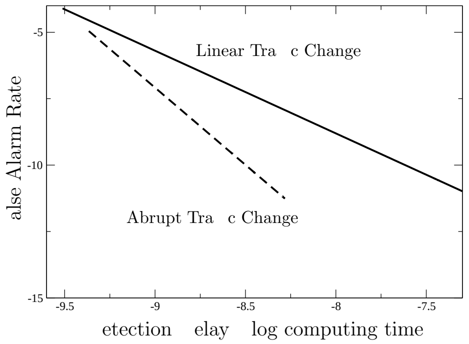

Fig. 4. A summary of the relationship between the detection delay and the false alarm rate for the SYN DoS attack (log–log scale).

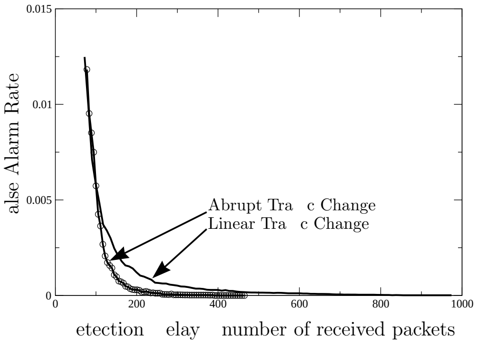

Fig. 5. A summary of the relationship between the log–detection delay and the false alarm rate for the SYN DoS attack.

V. DoS Attack Simulations

For simulations we have used a network simulator[1] ns with a network consisting of 100 nodes configured into a transit–stub topology which is depicted by Fig. 2. The network contained one transit domain, four transit nodes, and 12 stub domains with 96 nodes.

Under regular conditions, the traffic consisted of approximately 5% ICMP packets, 15–20% UDP packets, and 75–80% TCP packets. The attacker's activity represented less than 1% of traffic. After a 120 second period (measured using the simulator time) of regular traffic we have initiated one of the following three kinds of DoS attacks targeted at

[1] More information on the network simulator ns can be found on the internet at the address http://www.isi.edu/nsnam/ns/

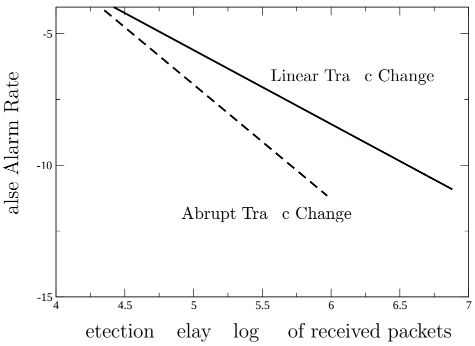

Fig. 6. A summary of the relationship between the detection delay and the false alarm rate for the SYN DoS attack (log–log scale).

the victim node: TCP SYN Flooding, UDP Packet Storm and ICMP Ping Flooding DoS attack (see [2], [3]).

During a DoS attack the attacker's traffic rapidly increased, reaching 20% of all traffic. We have considered two scenarios for the attacker's traffic increase: linear and abrupt. In the former case, the level of 20% of all traffic was reached in a linear manner during a 60 second interval, while in the latter situation the traffic increased to the 20% level immediately after the beginning of the attack.

During the simulations, we simultaneously observed all the statistics $N_{\text{pt}}^{k,i}$ and B_{SYN}^{k} with the length of the time intervals T_k set to 1 second. As shown in Fig. 1 for one particular run of a simulated UDP DoS, the sequential method has detected the attack in it's early stage. In this particular simulation of a UDP DoS attack with linear hostile traffic increase and threshold of 84.66., the detection delay was 35 simulated seconds.

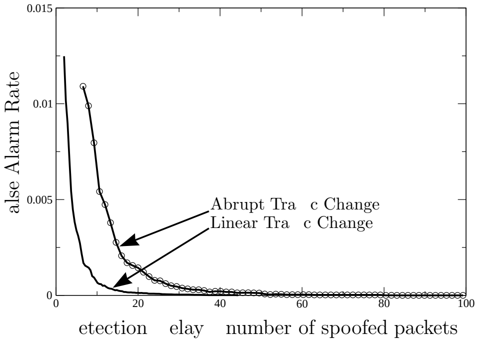

Fig. 7. A summary of the relationship between the detection delay and the false alarm rate for the SYN DoS attack.

It is of outmost importance to consider the detection delay in terms of the computational complexity and the computing time required for processing the observed data. Fig. 3 summarizes the detection delay (measured as the required computing time) in relation to the false alarm rate. The figure illustrates the performance of the sequential detection algorithm for both the linear and abrupt traffic increase scenarios of the SYN DoS attack. In the former case, the curve depicts the average detection delay based on 40 attack simulations, while in the latter situation the average detection delays are calculated using 23 simulations.

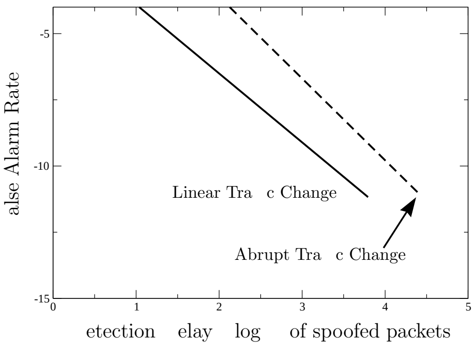

Fig. 8. A summary of the relationship between the detection delay and the false alarm rate for the SYN DoS attack (log–log scale).

TABLE I
Linear Traffic Increase.

False Alarm Rate	Detection delay measured using # of received packets	# of spoofed packets	Computing time (s)
0.0125	72.05	1.95	0.000074
0.0109	77.35	2.10	0.000078
0.0084	87.05	3.03	0.000083
0.0057	105.00	3.93	0.000095
0.0038	116.23	4.58	0.000101
0.0021	155.80	6.23	0.000128
0.00001	828.55	36.13	0.000577

TABLE II
Abrupt Traffic Increase.

False Alarm Rate	Detection delay measured using # of received packets	# of spoofed packets	Computing time (s)
0.0118	77.57	6.57	0.000086
0.0100	80.52	6.57	0.000087
0.0075	94.65	9.67	0.000100
0.0052	102.65	11.67	0.000103
0.0039	104.35	11.47	0.000103
0.0019	122.00	15.87	0.000108
0	375.87	77.00	0.000248

In addition, Fig. 5 and Fig. 7 depict the relationship of the false alarm rate and the detection delay in terms of the total number of observed SYN packets and in terms of the number of spoofed SYN packets. Fig. 4, 6, and 8 show smoothed versions of the corresponding performance indices of the detection procedure using the logarithmic scale. Tables I and II summarize the relationship of these three delay measures respectively for both scenarios: the linear and the abrupt change of the traffic pattern after the start of the attack.

VI. Conclusions

We proposed two adaptive DoS detection methods: purely sequential and batch–sequential. The detection algorithms are developed based on the change-point detection theory. Both methods belong to the class of Anomaly Detections Systems. They minimize the average delay of detection of an attack for a pre-specified rate of false alarms.

The results of simulations show that the algorithms work well in the sense that the simulated DoS attacks are detected in their early stages, well before the hostile traffic reaches it's full potential.

References

[1] J. Anděl, *Matematická Statistika*. SNTL – Nakladatelství Technické Literatury, Praha, the Czech Republic, 1985 (In Czech).

[2] CERT, *UDP Packet Storm*. CERT Advisory CA-1996-01, 1996.

[3] CERT, *TCP SYN Flooding and IP Spoofing Attacks*. CERT Advisory CA-96.21, 1996.

[4] H. Cramér, *Mathematical methods of statistics*. Princeton Univ. Press, 1946.

[5] M. Basseville and I.V. Nikiforov, *Detection of Abrupt Changes: Theory and Applications*. Prentice Hall, Englewood Cliffs, 1993.

[6] L. Gordon and M. Pollak, "Average run length to false alarm for surveillance schemes designed with partially specified pre-change distribution," *Ann. Statist.*, vol. 25, pp. 1284–1310, 1995.

[7] S. Kent, "On the trial of intrusions into information systems," *IEEE Spectrum*, pp. 52–56, December 2000.

[8] T.L. Lai, "Sequential changepoint detection in quality control and dynamical systems," *J. R. Statist. Soc.* B, vol. 57, No. 4, pp. 613–658, 1995.

[9] G. Lorden, "Procedures for reacting to a change in distribution," *Ann. Math. Statist.*, vol. 42, pp. 1987–1908, 1971.

[10] E.S. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, pp. 100–115, 1954.

[11] M. Pollak, " Optimal detection of a change in distribution," *Ann. Statist.*, vol. 13, pp. 206–227, 1986.

[12] S.L. Scott, "Detecting network intrusion using a Markov modulated nonhomogeneous Poisson process, *JASA*, (submitted for publication).

[13] A.N. Shiryaev, "On optimum methods in quickest detection problems," *Theory Probab. Appl.*, vol. 8, pp. 22–46, 1963.

[14] A.N. Shiryaev, *Optimal Stopping Rules*. Springer-Verlag, New York, 1978.

[15] D. Siegmund, *Sequential Analysis: Tests and Confidence Intervals*. Springer-Verlag, New York, 1985.

[16] A.G. Tartakovsky, *Sequential Methods in the Theory of Information Systems*. Radio i Svyaz', Moscow, 1991 (In Russian).

[17] A.G. Tartakovsky, "Efficiency of the Generalized Neyman–Pearson Test for Detecting Changes in a Multichannel System," *Problems of Information Transmission*, vol.28, pp.341-350, 1992.

[18] A.G. Tartakovsky and I.A. Ivanova, "Comparison of Some Sequential Rules for Detecting Changes in Distributions," *Problems of Information Transmission*, vol. 28, pp. 117-124, 1992.

[19] A.G. Tartakovsky, "Asymptotic Properties of CUSUM and Shiryaev's Procedures For Detecting a Change in a Nonhomogeneous Gaussian Process," *Mathematical Methods of Statistics*, vol. 4, No. 4, 1995.

[20] A.G. Tartakovsky, "Extended asymptotic optimality of certain change-point detection procedures: non-i.i.d. case," *Annals of Statistics* (submitted for publication).

ISBN 0-7803-9814-9/$10.00 ©2001 IEEE