The ISC Int'l Journal of Information Security

ISeCure

July 2009, Volume 1, Number 2 (pp. 71–90)

http://www.isecure-journal.org

Invited Paper

Moving Dispersion Method for Statistical Anomaly Detection in Intrusion Detection Systems

Jovan Dj. Golić [a,*]

[a] *Security Innovation, Telecom Italia, Via G. Reiss Romoli 274, 10148 Turin, Italy*

ARTICLE INFO.

Article history:
Received: 27 May 2009
Revised: 13 July 2009
Accepted: 15 July 2009
Published Online: 25 July 2009

Keywords:
Intrusion detection, Statistical anomaly detection, Dispersion measure, Concentration measure, Variance, Linear regression, EWMA techniques

ABSTRACT

A unified method for statistical anomaly detection in intrusion detection systems is theoretically introduced. It is based on estimating a dispersion measure of numerical or symbolic data on successive moving windows in time and finding the times when a relative change of the dispersion measure is significant. Appropriate dispersion measures, relative differences, moving windows, as well as techniques for their efficient estimation are proposed. In particular, the method can be used for detecting network traffic anomalies due to network failures and network attacks such as (distributed) denial of service attacks, scanning attacks, SPAM and SPIT attacks, and massive malicious software attacks.

1 Introduction

An *intrusion detection system (IDS)* aims at detecting and logging malicious, abusive, or suspicious processes, activities, or events in an information system such as a host computer or a system of computers communicating with each other via a communications network. In particular, a communications network, such as the Internet, may utilize the Internet Protocol (IP) as a communication protocol. A host-based IDS is running on a single host and mainly logs suspicious and unauthorized events or activities and changes to system files and configurations. A network-based IDS captures network packets on relevant network segments and inspects them. An intrusion prevention system (IPS) is in addition capable of taking an immediate protective action following intrusion detection. For example, a network-based IPS may drop malicious packets, block all further traffic from a particular IP address, or perform traffic shaping in terms of available transmission bandwidth. A network-based IDS/IPS has to deal with large volumes of data and hence has to be computationally very efficient. As the business cost of network problems may be high, especially for large networks (e.g., telephone operators, enterprises, etc.), it is practically very important to have effective solutions for IDS/IPS in these networks.

There are two main types of intrusion detection techniques: attack-based or attack detection techniques and anomaly-based or anomaly detection techniques. Attack-based (i.e., signature-based, pattern-based, or knowledge-based) techniques utilize a description (signature, pattern) of a concrete attack (e.g., virus, worm, or other malicious software) and decide if the observed data is consistent with this description or not; the attack is declared in the case of detected consistency. Anomaly-based or behavior-

* Corresponding author.
Email address: jovan.golic@telecomitalia.it (J. Dj. Golić).

ISSN: 2008-2045

based techniques utilize a description (profile) of normal/standard traffic or activities, rather than anomalous attack traffic or activities, and decide if the observed data is consistent with this description or not; an attack or anomalous traffic or activities are declared in the case of detected inconsistency. Both techniques have advantages and disadvantages.

Attack detection techniques require prior knowledge of each particular attack targeted and have to be continuously updated with new signatures corresponding to new attacks that appear. A signature can also describe a type of attacks instead of a particular attack. These techniques are in principle incapable of detecting previously unknown attacks, unless they can be recognized by the already existing signatures. However, they have zero false negative rates with respect to targeted attacks and typically very small false positive rates, the more so if the signatures used are more specific and hence closer to characterizing, not just describing the attacks. They are hence very reliable to be used for in-line intrusion prevention and especially effective against viruses, worms, Trojan, spyware, and other malicious software attacks. Of course, since malicious codes are subject to frequent mutation, the effective false negative rate will not be equal to zero when considered with respect to mutated attacks.

Anomaly detection techniques in principle do not require prior knowledge of particular attacks and as such are in principle capable of detecting previously unknown attacks. However, they typically have nonzero false negative rates, with respect to given attacks or types of attacks, and higher false positive rates in a sense that they can declare anomalous traffic or activities in the absence of attacks. Therefore, they are typically not reliable enough to be used for in-line intrusion prevention, but are useful as complementary tools, in addition to attack-based techniques, for detecting anomalous traffic or computer activities, possibly due to network failures or new, previously unknown attacks. They can essentially be classified into two broad categories: rule-based techniques and statistic-based or statistical techniques. In particular, they may be useful against broad classes of network attacks such as (distributed) denial of service attacks ((D)DoS attacks), scanning or probing attacks (e.g., port-scanning attacks), SPAM and SPIT (SPam over Internet Telephony) attacks, as well as worm or virus outbreaks and other massive malicious software attacks.

There are really numerous publications, in terms of refereed papers and patents, proposing a wide range of various statistical methods that can be used for modeling normal network traffic or computer activities and for deciding if the observed data is consistent with these models. For example, for data clustering, neural network, and principal component analysis techniques, see [1], [2], and [3], respectively, and for discrete wavelet transforms and waveform correlation techniques, see [4] and [5]. As for the relevant network data features, it is suggested in [6] that the curves representing the packet rate, byte rate, and flow rate (i.e., the number of packets, bytes, and flows per second, respectively) in time can be useful for detecting and classifying network traffic anomalies, possibly through the wavelet transform techniques. On the other hand, a use of mean values and variances, dynamically estimated by the Exponentially Weighted Moving Average (EWMA) technique on historical or past data, together with the outlier classification principle, is proposed in [7], [8], [9], and [10], whereas some methods for monitoring the variance are proposed in [7] and [11]. A use of the Shannon entropy is proposed in [12], [3], and [13], while lossless data compression algorithms are employed in [14]. The chi-square statistic with respect to a baseline profile distribution computed on past or historical data is applied in [12].

The main objective of this paper is to provide a new theoretical framework for statistical anomaly detection in intrusion detection systems with a relatively low complexity suitable for applications in high-speed and high-volume communications networks. The new general method is based on detecting abrupt relative changes in a dispersion measure associated with a monitored data stream up to a current point in time. The underlying assumption is that for many types of network and computer attacks, the dispersion of appropriately chosen data features is expected to significantly increase or decrease in the presence of the attack. The main contributions of this paper are summarized below.

Relative changes are measured by appropriately defined relative differences between the current and preceding values of the dispersion measure, e.g., the relative squared differences. For stationary data, the differences should be relatively small and roughly insensitive to data, so that possibly dynamic thresholds should reflect only the nonstationary nature of data.

For numerical data, such as the packet or byte rate and average packet size for network traffic, the proposed dispersion measure is either the variance, which can be regarded as the least mean squared error resulting from a constant approximation of data, or, more generally, the linear least square error (LLSE) as the least mean squared error resulting from a linear (more precisely, affine) approximation of data which can be obtained by the linear regression technique. The LLSE is less sensitive to changes in nonstation-

ary or correlated normal data than the variance. Note that even the proposed relative squared difference of variances associated with successive moving windows of data in time appears to be a novel statistical anomaly detection method, since the previous work uses the variance in different ways, e.g., for outlier classification or for comparison with reference values. For multidimensional, possibly correlated numerical data, the relative squared difference of variances is generalized in terms of the associated covariance matrices.

For symbolic data, such as the IP addresses, port numbers, email addresses, and Universal Resource Identifiers (URIs) in network traffic, it is proposed to use the so-called quadratic entropy associated with the relative frequencies of symbolic values as the main dispersion measure. Equivalently, one can use the corresponding quadratic concentration measure or the chi-square statistic with respect to uniformly distributed symbolic values. The Shannon entropy, which can be interpreted in terms of data compression coding, is less suitable as a dispersion measure, because it is overly sensitive to changes in less frequent symbolic data. If the number of observed samples is much smaller than the number of possible symbolic values, then the number of repetitions among the observed values is suggested as another concentration measure applicable.

In addition, for both numerical and symbolic data, the average conditional dispersion measures are introduced and argued to be especially effective for detecting certain general types of network attacks. They are conditioned on the values of certain symbolic variables such as, e.g., the source or destination IP addresses, email addresses, or URIs.

For both numerical and symbolic data, three types of moving windows in time are proposed for the computation of dispersion measures, namely, a single sliding window of a finite length, a pair of sliding windows of finite lengths where a longer one is composed of a shorter one and appended current data, and a moving window of increasing length extending from the initial to the current time. For the first and third types, the dispersion measure is compared for two consecutive moving windows, at the current and preceding times, whereas for the second type, the dispersion measure is compared for the two sliding windows at each time. For the third type of moving windows, adapted EWMA techniques are developed for all the dispersion measures, namely, two EWMA techniques for the variance, novel EWMA techniques for the LLSE measure and the quadratic concentration measure, as well as novel EWMA techniques for average conditional dispersion and concentration measures and for multidimensional data.

Iterative techniques for the efficient computation of the dispersion and concentration measures are derived for all types of windows considered. The EWMA techniques are better suited to nonstationary normal data and enable a significant reduction of the amount of data needed to be memorized, in comparison with the sliding window techniques.

The rest of the paper is organized as follows. Section 2 contains a brief overview of anomaly detection techniques and related attacks to which they may be applicable. The basic numerical and symbolic data features for network-based and host-based intrusion detection are outlined in Section 3. A more detailed discussion of previous work in view of the main lines of the new statistical anomaly detection approach is presented in Section 4. The three types of moving windows utilized are described in more detail in Section 5 and the new methods for statistical anomaly detection are presented in Sections 6 and 7, for numerical features, and in Section 8, for symbolic features. Network-based intrusion detection applications are discussed in more detail in Section 9. Conclusions are presented in Section 10 and iterative update expressions for the sliding window techniques are given in the Appendix.

2 Anomaly Detection Techniques and Related Attacks

Rule-based anomaly detection techniques describe the normal behavior in terms of certain static rules or certain logic and can essentially be stateless or stateful. In particular, such rules can be derived from protocol specifications. An important class of these techniques in fact performs stateless or stateful protocol analysis and is useful for detecting malformed or invalid messages (packets or sequences of packets), which may appear during the attacks such as DoS attacks. So, if the rules are violated, then anomalous behavior is declared. The false positive rate is thus equal to zero, as the normal behavior has to satisfy the rules chosen. Of course, if practical implementations deviate from the specified rules, then the false positive rate will not be zero. The false negative rate for a given attack or a given anomalous behavior is usually nonzero, as the rules chosen may be satisfied even if the behavior is anomalous, and the more so if the rules are less specific.

Statistical anomaly detection techniques describe the normal behavior in terms of the probability distributions of certain variables, called statistics, depending on the selected data features. This can possibly be achieved by using appropriate statistical models of

normal network traffic or computer activities on historical or past data. Since the normal behavior is typically nonstationary, these models may be dynamic, e.g., may depend on a time of the day or a day in the week or may be adaptive. In a comparison stage, a data portion of the current traffic or computer activities (e.g., a single packet or a sequence of packets of network traffic) is then compared with the developed statistical model and a decision is made whether the deviation from the statistical model is statistically significant or not. If yes, then the considered portion of the current data is classified as anomalous and an alert is issued.

A single observed value of a chosen statistic can be compared with the statistical model of normal behavior by direct checking if the value belongs to a range of normal values, and it is declared anomalous if it falls out of this range. More generally, for a sequence of observed values, a measure of deviation from or correlation with the statistical model can first be computed and, then, it is checked if it belongs to a range of normal values. Typical deviation measures include the chi-square statistic and the Kullback-Leibler divergence between the observed and expected probability distributions, correlation measures between sequences, and prediction and residual mean squared errors.

The range of normal values is usually specified in terms of thresholds, which are chosen according to an acceptable false positive rate. If this rate is chosen to be very small, then the false negative rate with respect to a given attack or type of attacks may become unacceptably high, so that a tradeoff between the two rates is necessary. If a statistic inherently does not reflect the attack behavior sufficiently accurately, then the false negative rate will be high almost regardless of the thresholds chosen and the statistic is hence not very useful. On the other hand, if the statistical model does not reflect dynamic behavior of normal data, then the false positive rate may be high. Therefore, the thresholds may be chosen dynamically or even adaptively. However, if the attacker knows the adaptation policy, then the attack data can also be adapted accordingly, in order to satisfy the dynamic thresholds.

The main types of network or computer attacks where statistical anomaly detection techniques are useful are described in the sequel.

DoS attacks and distributed DoS (DDoS) attacks are commonly regarded as a major threat to the Internet. A DoS attack is an attack on a computer system or network that causes a loss of service or network connectivity to legitimate users, that is, unavailability of services. Most common DoS attacks aim at exhausting the computational resources, such as connection bandwidth, memory space, or CPU time, for example, by flooding the target network node by valid or invalid requests and/or messages. They can also cause disruption of network components or disruption of configuration information, such as routing information, or can aim at disabling an application making it unusable. In particular, the network components (e.g., servers, proxies, gateways, routers, switches, hubs, etc.) may be disrupted by malicious software attacks, for example, by exploiting buffer overflows or vulnerabilities of the underlying operating system or firmware.

A DDoS attack is a DoS attack that, instead of using a single computer as a base of attack, uses multiple compromised computers simultaneously, possibly a large or a very large number of them (e.g., millions), thus amplifying the effect. Altogether, they flood the network with an overwhelming number of packets which exhaust the network or application resources. In particular, the packets may be targeting one particular network node causing it to crash, reboot, or exhaust the computational resources. The compromised computers, which are called zombies or bots, are typically infected by malicious software (worm, virus, or Trojan) in a preliminary stage of the attack, which involves scanning a large number of computers searching for those vulnerable. The attack itself is then launched at a later time, either automatically or by a direct action of the attacker. (D)DoS attacks are especially dangerous for Voice over IP (VoIP) applications, e.g., based on the Session Initiation Protocol (SIP). In particular, the underlying SIP network dealing only with SIP signalling packets is potentially vulnerable to request or message flooding attacks, spoofed SIP messages, malformed SIP messages, and reflection DDoS attacks. Reflection DDoS attacks work by generating fake SIP requests with a spoofed source IP address and a spoofed via header field, which falsely identify a victim node as the sender, and by sending or multicasting them to a large number of SIP network nodes, which all respond to the victim node, and repeatedly so if they do not get a reply, hence achieving an amplification effect.

SPAM attacks consist in sending unsolicited electronic messages (e.g., through E-mail over the Internet), with commercial or other content, to numerous indiscriminate recipients. Analogously, SPIT attacks consist in sending SPAM voice messages in VoIP networks. *Malicious software attacks* consist in sending malicious software, such as viruses, worms, Trojan, or spyware, to numerous indiscriminate recipients, usually in a covert manner. *Scanning or probing attacks* over the Internet consist in sending request messages

in large quantities to numerous indiscriminate recipients and to collect the information from the provoked response messages, particularly, in order to detect vulnerabilities to be used in subsequent attacks. For example, in port scanning attacks, the collected information consists of the port numbers used by the recipients. Since DDoS attacks are typically preceded by (massive) scanning attacks, detecting the scanning attacks may also help prevent DDoS attacks.

3 Data Features

The data features to be used for network-based intrusion detection are extracted from network traffic containing data packets such as IP packets in IP-based networks. The main IP packet header information contained in layers 3 and 4 (the network and transport layers, respectively) includes source IP address, TCP/UDP source port number, destination IP address, TCP/UDP destination port number, and transport protocol used. A series of packets having in common this basic information is commonly defined as a flow. A basic feature of an IP packet is its size in bytes, i.e., the total number of layer 3 bytes in a packet. With respect to a reference short time interval of length/duration ΔT, the basic numerical features regarding a flow include the packet rate (i.e., the number of packets per second) $R_{packet} = N_{packet}/\Delta T$, the byte rate (i.e., the number of layer 3 bytes per second) $R_{byte} = N_{byte}/\Delta T$, and, if $N_{packet} > 0$, the average packet size, in bytes, $P_{size} = R_{byte}/R_{packet}$. They can be traced in time by looking at successive short time intervals, where the length ΔT specifies the time resolution with which the traffic is monitored and can be static or dynamic. The starting and ending times of the first and the last packet in a flow, respectively, as well as the total number of monitored flows can also be reported.

These numerical features can be traced in time at a chosen network node or a set of nodes, for each individual flow along a communication link or for selected flows simultaneously, e.g., chosen according to the IP addresses or port numbers in order of decreasing packet rate. The numerical features for individual flows can also be aggregated according to selected packet parameters such as the symbolic packet features including the source or destination IP addresses, the source or destination port numbers, and the transport protocol used. For example, the flows with the same source or destination IP address can be grouped together. The aggregated packet features then correspond to the outbound or inbound traffic for a particular network node, respectively. The number of simultaneously monitored flows with the same source/destination IP address is another numerical feature indicating the activity of a particular node. On the other hand, aggregation of flows according to port numbers is indicative of the applications of the packets transmitted. If the features are grouped for all the flows monitored, where the direction of flows along a communication link can possibly be distinguished, then anomaly detection relates to the network traffic as a whole.

Other symbolic packet features of interest include information extracted from other packet header layers such as the application layer, e.g., the source and destination email addresses, the source and destination SIP URIs, the HTTP (Hypertext Transfer Protocol on the World Wide Web) URIs, or the type of SIP packets transmitted in a VoIP network. In general, for privacy and anonymization issues, the information extracted from the packet headers is more interesting for the network traffic analysis than the content-related information from the packet payloads. The symbolic packet features can be used by themselves or in connection with the numerical packet features, as described above. Any numerical data can also be regarded as symbolic, possibly after appropriate quantization, but not the other way around.

The data features to be used for host-based intrusion detection are extracted from audit trail data capturing in time the activities on a host machine in terms of the security-relevant events. Different types of these events are then the basic symbolic data features monitored in time. With respect to a reference short time interval of length ΔT, the basic numerical feature is then the so-called event intensity [7] defined as the number of events of a given type that occurred in this interval divided by ΔT. The event intensity can be aggregated according to specified types of events. It corresponds to the packet rate extracted from network traffic data.

4 Related Previous Work

In view of the nonstationary nature of network traffic data, many previous papers or patents on statistical anomaly detection for network-based intrusion detection propose a dynamic estimation of reference statistical models and probability distributions on historical data collected from normal traffic in communications networks, usually depending on a time of the day or a day in the week. Alternatively, the statistical models can be derived adaptively from the past data preceding a current portion of network traffic or computer activities to be classified as normal or anomalous. In both cases, the current portion of network traffic or computer audit data under consideration is checked for consistency with the derived statistical model.

Another paradigm present in some previous work is to look for sudden changes in certain numerical features derived from the original data, e.g., in the traffic volume measured by the byte or packet rate. In this paper, the derived numerical features are related to statistical models, so that the applied paradigm is essentially to look for sudden changes in the underlying statistical model, where the estimated statistics relate to various dispersion or concentration measures associated with moving windows of traffic or audit data and the changes are measured by appropriately defined relative differences. The original data monitored and analyzed may be of numerical or symbolic nature. Numerical data can be expressed in terms of integer, rational, or real numbers, so that one can measure the distance or closeness between two data values by the Euclidean metric. For symbolic data, the distance or closeness between two data values cannot be defined or is not considered to be relevant or meaningful. The set of values for symbolic data is usually finite or countably infinite. The proposed dispersion measures are essentially based on the Euclidean metric for numerical data and on the relative frequencies for symbolic data.

The origins of the new unified method in previous work are pointed out in more detail in the sequel. Like in other areas of statistical anomaly detection, the conceptual and/or technical differences among various methods and techniques proposed are sometimes very subtle.

In [6], it is suggested that packet rate, byte rate, and flow rate (i.e., the number of packets, bytes, and flows per second) curves in time can be useful for detecting and classifying traffic anomalies, possibly through the wavelet transform techniques.

In [9], a method is described wherein the packet and byte rates are considered as functions of time and, at each time, the mean values and variances of these rates are estimated by using historical data, possibly by the EWMA technique, and then a given sample of traffic at a given time is classified by comparing its packet and byte rates with a threshold being proportional to the sum of the historical mean value and the historical standard deviation (i.e., the square root of the variance) multiplied by a positive constant. Anomalous traffic is declared if the threshold is exceeded, i.e., if the observed sample of traffic is classified as an outlier. A similar method where the mean value and the variance are estimated as the EWMAs, with different, but mutually related associated constants is disclosed in [8].

In [7], it is proposed to apply the EWMA techniques for dynamically estimating the mean values and variances of the event intensity process derived from the audit trail data describing the activities on a host machine in a computer network. Anomaly detection is based on the outlier classification principle, where the thresholds are determined under certain probabilistic models for the event intensity process. Alternatively, anomaly detection is based on the estimated variance only, which is compared with a reference value and an alert is then declared if the ratio of the two values is too large or too small. In [10], a similar EWMA technique together with the outlier classification principle is applied to the alert intensity process in order to distinguish outstanding alerts.

A flow identification method [15] for VoIP media traffic uses the flow statistics such as the minimal and maximal values of the packet inter-arrival time and some characteristics of the packet size distribution comprising the minimal, maximal, average, and median values as well as the total number of different packet sizes occurring in a flow. The statistics are calculated and compared with reference patterns on short time intervals (e.g., 1 second long) and the verification results are averaged over a longer time interval in order to classify a given flow.

A method [11] relating to VoIP data traffic consists of computing the empirical variance estimates of the normalized byte rate on overlapping sliding windows and comparing them with predicted variances that are theoretically obtained under probabilistic models for the number of calls per second. At any time, an anomaly is declared if the ratio of the empirical and theoretical variances is greater than a threshold, which falls in the range between one and two.

In [5], it is proposed to dynamically apply a discrete wavelet transform to overlapping sliding windows of the byte rate curves in time and to look for sudden changes in the logarithms of the associated energy distribution coefficients in order to detect DDoS attacks.

A technique [16] for detecting (D)DoS attacks using randomly spoofed IP addresses consists of counting the relative number of different values of hashed IP addresses among a number of packets, which are inspected successively in time, and of comparing this number with a predetermined threshold. A (D)DoS attack is declared if the threshold is exceeded. Hashing serves for reducing the number of values. The number of inspected packets is iteratively increased if an attack is not detected.

A technique according to which DDoS attacks can be (proactively) detected even near the sources of the attack by checking for an increase of new source IP addresses appearing, provided that the source IP addresses of the attack traffic are randomly spoofed, is

proposed in [17]. It should be noticed that according to this article the IP addresses are monitored in non-overlapping time intervals and the increase is measured with respect to a database of legitimate IP addresses collected during off-line training.

In [12], two methods for DDoS attack detection based on discrete packet features are proposed. One method uses the Shannon entropy of source IP addresses estimated on sliding windows consisting of a number (e.g., 10,000) of packets and looks for times when the entropy exceeds a predefined threshold. The other method applies the chi-square statistic of a current absolute frequency distribution with respect to a baseline profile distribution as the expected distribution, where the discrete values can be grouped in bins, and looks for times when the statistic exceeds a threshold. The absolute frequencies are multiplied by the exponentially decaying aging factors, where the baseline profile half-life is much longer than the current profile half-life. The baseline profile can be estimated on past data or historical data (e.g., for the same daily period).

A technique proposed in [3] consists of estimating the Shannon entropy of discrete packet features such as IP addresses and port numbers, in non-overlapping, relatively short time intervals (e.g., 5 min), statistically modeling the multidimensional entropy data collected on multiple links in a communications network by using the principal component analysis, and then verifying if the current data is inconsistent with the model by checking if the residual mean squared error with respect to the subspace identified by the principal component analysis exceeds a threshold. This comparison technique is not applicable to one-dimensional data, as in this case there are no residual errors. It is expected that the frequency distribution of the IP addresses or port numbers reflected in the normalized Shannon sample entropy should change in the case of an attack traffic. A disadvantage of the Shannon entropy is that it is very sensitive to changes in very low frequencies of symbolic data. Anomalies are then classified by applying the cluster analysis to the residual entropy vectors.

A technique applied in [13] also uses the Shannon entropy estimates of discrete packet features obtained at multiple nodes in a communications network (e.g., points of presence), and then essentially combines them in a joint entropy estimate, where the relative frequencies associated with the nodes correspond to the traffic volumes expressed in terms of a number of packets. DDoS attack detection is then performed by applying the outlier classification principle to the joint entropy estimates obtained on sliding windows.

A method [14] considers discrete packet features such as IP addresses in relatively short time intervals (e.g., 5 min) and compresses a concatenation of all the IP addresses occurring in an interval by a lossless data compression algorithm, such as the well-known Lempel-Ziv coding algorithm. It is expected that the compression ratio should be lower if there is a massive worm attack traffic in the interval, due to randomization of destination IP addresses. However, it is not specified how to measure if the decrease is significant or not. Apart from the computational and memory issues, a disadvantage of Lempel-Ziv-based compression algorithms in comparison with the Shannon entropy is that they are not invariant under the order of the symbolic data.

A number of packet statistics for the detection of DDoS attacks are suggested in [18] and [19]. The statistics examined in [18] include the number of open or half-open (obtained from TCP flags) connections, the number of transmitted or received bytes per (grouped) IP address, the number of open ports per (grouped) IP address, and the histogram of the average packet sizes, while the statistics examined in [19] include the histogram of the flow sizes in bytes over a time period and the activity of (grouped) IP addresses.

5 Moving Windows

Let x be a generic numerical or symbolic feature extracted from network traffic or computer audit data in elementary short time intervals of length ΔT, where ΔT can vary in time (e.g., this occurs if symbolic feature samples are taken from individual packets of network traffic). Let $X = (x_i)_{i=1}^{\infty}$ denote the corresponding sequence of samples of the feature x taken in time. Sections 6 and 7 deal with numerical features, whereas Section 8 deals with symbolic features.

The essence of the moving dispersion method is to compute successively a chosen dispersion measure at chosen discrete times and to look for points in time when the values of the dispersion measure change suddenly, where the changes are measured by relative rather than absolute differences. Three types of moving windows in time are proposed for the computation of dispersion measures, namely, a single sliding window of a finite length, a pair of sliding windows of finite lengths where a longer one is composed of a shorter one and appended current data, and a moving window of increasing length extending from the initial to the current time. For the first and third types, the dispersion measure is compared for two consecutive moving windows, at the current and preceding times, whereas for the second type, the dispersion measure is compared for the two sliding windows at each time.

A single sliding window of length T is sliding in time, starting from an initial position, each time advancing τ units of time, where T and τ are fixed parameters. If ΔT is static, i.e., fixed, then T and τ are defined as fixed integer multiples of ΔT. If ΔT is dynamic, i.e., variable in time, then it is assumed that the sliding window at each time contains an integer number of elementary time intervals and approximately has the same length T. Accordingly, two consecutive windows of (approximately) the same length T are shifted τ units of time from each other and hence overlap over $T-\tau$ units of time. In a general case, when the samples of x are taken possibly irregularly in time, i.e., in elementary time intervals of possibly variable length ΔT, the number of samples per sliding window may vary in time, and so may the numbers of overlapping and non-overlapping samples in two consecutive sliding windows. At each time, the dispersion measure is computed and compared for two consecutive sliding windows, at the current and preceding times.

The value of τ determines the resolution of the proposed statistical anomaly detection method, because it takes τ units of time, or a small multiple of them, in order to detect a change from normal to anomalous data (or vice versa). The value of T should be large enough in order to obtain relatively stable estimates of the chosen dispersion measure, so that the relative changes of the dispersion measure are not too large for normal data. For the same reason, in view of the nonstationary nature of normal data, T should not be too large. The ratio T/τ should not be too large so that the change of data from normal to anomalous would not require a very small threshold to be detected. For example, one may take $1 \le T/\tau \le 10$.

For the second type of moving windows, apart from a longer sliding window of length T described above, another, shortened sliding window of length $T-\tau$ is also defined. Both windows are sliding in time, each time advancing τ units of time. At each time, the dispersion measure is computed and compared for the two sliding windows, i.e., the longer one at the current time and the shorter one at the preceding time. Accordingly, unlike the single sliding window technique, the past data leaving the current sliding window of length T are thus excluded from the comparison. This way the sensitivity to a change of the dispersion measure is much higher when the change occurs near the ending points than around the starting points of the windows, which is not the case with the single sliding window technique, where there is a symmetry between the two sensitivities. Therefore, the sliding window pair technique may be more suitable for very nonstationary normal data or for detecting anomalous data of very short duration, especially if the window size T is relatively large.

The third type of moving windows having increasing length extend from a fixed initial time up to the current time. Their starting point is thus fixed, whereas the ending point each time advances τ units of time. They are suited for estimating the dispersion measure by appropriately defined EWMAs, so that the influence of the past data on the current dispersion measure diminishes with time, in order to ensure the sensitivity to anomalous behavior of the current data. This is especially important if the normal data is not stationary. At each time, the dispersion measure is computed and compared for two consecutive moving windows, at the current and preceding times.

Let $X_{m_j-n_j+1}^{m_j} = (x_i)_{i=m_j-n_j+1}^{m_j}$ denote the segment of samples corresponding to a generic j-th sliding window of length T, for $j \ge 1$, where, initially, $m_1 = n_1$. If ΔT varies in time, then the number of samples n_j in a segment is in general variable and so is $m_j - m_{j-1}$. If ΔT is fixed, then both n_j and $m_j - m_{j-1}$ are fixed. For the second type of moving windows, the segment of samples corresponding to a shortened $(j-1)$-th sliding window of length $T-\tau$ is then $X_{m_j-n_j+1}^{m_{j-1}} = (x_i)_{i=m_j-n_j+1}^{m_{j-1}}$. Finally, for the third type of moving windows, the segment of samples corresponding to a generic j-th moving window is then $X_1^{m_j} = (x_i)_{i=1}^{m_j}$.

Let $X(j)$ and $\acute{X}(j-1)$ denote segments of samples that are compared with each other at the j-th discrete time $t = m_j$. Accordingly, we have $X(j) = X_{m_j-n_j+1}^{m_j}$ and $\acute{X}(j-1) = X(j-1) = X_{m_{j-1}-n_{j-1}+1}^{m_{j-1}}$, $X(j) = X_{m_j-n_j+1}^{m_j}$ and $\acute{X}(j-1) = \hat{X}(j-1) = X_{m_j-n_j+1}^{m_{j-1}}$, and $X(j) = X_1^{m_j}$ and $\acute{X}(j-1) = X(j-1) = X_1^{m_{j-1}}$, for the three considered types of moving windows, respectively.

6 Moving Variance Method for Numerical Features

For numerical features, the first dispersion measure proposed is the variance of numerical samples in a considered moving window. Let $\sigma^2(j)$ and $\acute{\sigma}^2(j-1)$ denote the respective estimates of variance associated with the segments of samples $X(j)$ and $\acute{X}(j-1)$ to be compared with each other at the j-th discrete time $t = m_j$. A relative difference of variances is then defined as the relative squared difference

$$\delta_j = \frac{(\sigma^2(j) - \acute{\sigma}^2(j-1))^2}{\sigma^2(j)\acute{\sigma}^2(j-1)} = \frac{\sigma^2(j)}{\acute{\sigma}^2(j-1)} + \frac{\acute{\sigma}^2(j-1)}{\sigma^2(j)} - 2. \quad (1)$$

The singularities are formally treated as follows. If both the values $\sigma^2(j)$ and $\acute{\sigma}^2(j-1)$ are equal to 0, then $\delta_j = 0$, and if only one of them is equal to zero, then $\delta_j = \infty$. The relative squared difference is then compared with a fixed or dynamic threshold θ_j and if the threshold is exceeded once or a specified number of times in a row, then an alert for anomalous data is generated. An interesting feature of (1) is that the relative squared difference only depends on the ratio of the two variances and has the same value regardless of whether the variance increases or decreases.

Since the threshold relates to relative instead of absolute changes in variance, it may even be fixed and independent of the variance of normal data, provided that the data is stationary. More precisely, if the samples are drawn independently according to the same probability distribution, then, even if the two segments $X(j)$ and $\acute{X}(j-1)$ are not overlapping (i.e., if $\tau = T$ for the single sliding window technique), the relative squared difference of variances is generally a small random variable inversely proportional to the (effective) number of samples per segment and does not depend on the variance of the samples. For nonstationary normal data, a fixed threshold may be empirically determined depending on the false positive rate to be achieved. For the sliding window techniques, the threshold may generally increase as T/τ decreases, because the sensitivity to changes in variance is then increased.

Alternatively, to account for possibly considerable changes in variance of nonstationary normal data, the threshold could be dynamic and empirically determined from historical data or from past data adaptively, in order to keep the false positive rate reasonably low. In particular, it may depend on the time of the day for network traffic. Note that, by definition, the method is much more sensitive to changes in the variance than in the mean value, which may be considerable for typical normal data.

6.1 Sliding Window Techniques

For the sliding window techniques, the variance estimate can be computed as

$$\sigma^2(j) = \frac{1}{n_j} \sum_{i=m_j-n_j+1}^{m_j} (x_i - \mu(j))^2, \tag{2}$$

where $\mu(j) = \left(\sum_{i=m_j-n_j+1}^{m_j} x_i\right)/n_j$ is the mean value estimate. For unbiased estimates of variance, one may divide by $n_j - 1$ instead of n_j, but the numerical impact on the method would be negligible. For a single sliding window, $\hat{\sigma}^2(j-1) = \sigma^2(j-1)$. For a pair of sliding windows, $\acute{\sigma}^2(j-1) = \hat{\sigma}^2(j-1)$, where

$$\hat{\sigma}^2(j-1) = \frac{1}{\hat{n}_{j-1}} \sum_{i=m_j-n_j+1}^{m_{j-1}} (x_i - \hat{\mu}(j-1))^2, \tag{3}$$

where $\hat{n}_{j-1} = n_j + m_{j-1} - m_j$ is the number of samples in $\hat{X}(j-1)$ and $\hat{\mu}(j-1) = \left(\sum_{i=m_j-n_j+1}^{m_{j-1}} x_i\right)/\hat{n}_{j-1}$ is the corresponding mean value estimate. Instead of computing (2) and (3) for each j, which is not efficient if the underlying segments contain a lot of samples in common, one may use iterative update expressions given in the Appendix.

6.2 EWMA Techniques

For the third type of moving windows, the variance can be estimated iteratively by the standard EWMA technique, used in [7]. Let μ_k and σ_k^2 denote the estimated mean value and variance for a generic segment of samples $X_1^k = (x_i)_{i=1}^k$, $k \geq 1$. Given two constants α and β such that $0 < \alpha, \beta \leq 1$, and starting from $\sigma^2(x) = \overline{(x-\overline{x})^2}$, μ_k and σ_k^2 are iteratively computed as

$$\mu_k = \beta x_k + (1-\beta)\mu_{k-1} \tag{4}$$

$$\sigma_k^2 = \alpha(x_k - \mu_k)^2 + (1-\alpha)\sigma_{k-1}^2 \tag{5}$$

with the inital values $\mu_1 = x_1$ and $\sigma_1^2 = 0$. In particular, we may have $\alpha = \beta$. Explicit solutions are then given as

$$\mu_k = (1-\beta)^{k-1}x_1 + \beta\sum_{i=2}^{k}(1-\beta)^{k-i}x_i \tag{6}$$

$$\sigma_k^2 = \alpha\sum_{i=2}^{k}(1-\alpha)^{k-i}(x_i - \mu_i)^2. \tag{7}$$

Accordingly, at time $t = k$, this variance estimate measures the exponentially weighted average deviation of the initial k data samples from the corresponding mean values at the same times. The relative squared difference at time $t = m_j$ is then computed by (1) with $\sigma^2(j) = \sigma_{m_j}^2$ and $\acute{\sigma}^2(j-1) = \sigma^2(j-1) = \sigma_{m_{j-1}}^2$.

Alternatively, starting from $\sigma^2(x) = \overline{x^2} - (\overline{x})^2$, σ_k^2 can be iteratively computed by another technique, used in [10], in which the EWMA recursion is proposed to be applied to the mean squared values, i.e., by

$$\xi_k^2 = \alpha(x_k)^2 + (1-\alpha)\xi_{k-1}^2 \tag{8}$$

$$\sigma_k^2 = \xi_k^2 - (\mu_k)^2 \tag{9}$$

with the inital values $\mu_1 = x_1$ and $\xi_1^2 = (x_1)^2$. If $\alpha = \beta$, then the corresponding explicit solution for the variance is then given as

$$\sigma_k^2 = \alpha \sum_{i=2}^{k} (1-\alpha)^{k-i}(x_i - \mu_k)^2, \tag{10}$$

which, at time $t = k$, measures the exponentially weighted average deviation of the initial k data samples from the overall mean value up to the time $t = k$. The alternative technique is hence more sensitive for detecting anomalous changes in data, at the expense of somewhat larger variations of variance estimates in normal data.

The constants α and β determine the effective number of past samples influencing the variance and mean value estimates, respectively. An equivalent number of samples corresponding to α is given as $n = 2/\alpha - 1$ and hence relates to an equivalent size T of the corresponding sliding window. In standard applications of the EWMA technique, e.g., for detecting data outliers or for reducing the noise in data by smoothing, the used constants are typically close to 1 or at least moderately large. In the moving variance method, they should be relatively small and, like $T/\Delta T$ in the sliding window techniques, empirically adapted to the statistical properties of normal data. In general, bigger constants or, equivalently, smaller values of $T/\Delta T$ should correspond to faster variance variations in normal data.

6.3 Average Conditional Variance

Let a numerical feature x be associated with a symbolic feature s, i.e., let us consider a two-dimensional feature (x, s). For example, for network traffic data, x can be the byte rate, packet rate, or average packet size and s can be the IP address or port number. Instead of considering the variance associated with the aggregated numerical feature x, as described above, it may be advantageous to consider the conditional variance of x, conditioned on s, and then averaged over s. Let $\sigma^2(x|s)$ denote the variance of x conditioned on a concrete value of s and let $\overline{\sigma^2(x|s)}$ denote the corresponding average over s which is called the average conditional variance. Let $\overline{x|s}$ denote the mean value of x conditioned on s. We then have

$$\overline{\sigma^2(x|s)} = \overline{\left(x - \overline{x|s}\right)^2} = \overline{x^2} - \overline{\left(\overline{x|s}\right)^2}. \tag{11}$$

The method then goes along the same lines as above, but with the average conditional variance instead of the variance.

The sequence of samples is now $(X, S) = (x_i, s_i)_{i=1}^{\infty}$. For the sliding window techniques, let $n_{j,s}$ denote the number of samples in the segment $X(j)$ such that $s_i = s$ and let

$$\mu_s(j) = \frac{1}{n_{j,s}} \sum_{\substack{i=m_j-n_j+1 \\ s_i=s}}^{m_j} x_i \tag{12}$$

denote the estimated mean value conditioned on s. The average conditional variance estimate can then be computed as

$$\overline{\sigma^2}(j) = \frac{1}{n_j} \sum_{i=m_j-n_j+1}^{m_j} (x_i - \mu_{s_i}(j))^2, \tag{13}$$

and $\overline{\hat{\sigma}^2}(j-1)$ is computed analogously.

The two EWMA techniques from Section 6.2 can be adapted to deal with average conditional variances, by using the two equivalent expressions from (11) and by applying the EWMA recursions to the involved mean values, respectively. For both the techniques, we thus need to compute recursively the conditional mean values $\overline{x|s}$ for each value of s. Since each particular value of s comes irregularly in time, the corresponding EWMA recursion of the type (4) is updated at irregular times, so that at time $t = k$, only the mean value conditioned on $s = s_k$, denoted as μ_{k,s_k}, is updated. The EWMA recursion corresponding to the left-hand expression in (11), $\overline{\left(x - \overline{x|s}\right)^2}$, is then

$$\overline{\sigma}_k^2 = \alpha(x_k - \mu_{k,s_k})^2 + (1-\alpha)\overline{\sigma}_{k-1}^2. \tag{14}$$

On the other hand, to the right-hand expression in (11), $\overline{x^2} - \overline{\left(\overline{x|s}\right)^2}$, there correspond two recursions, i.e., (8) and

$$\overline{\mu}_k^2 = \alpha(\mu_{k,s_k})^2 + (1-\alpha)\overline{\mu}_{k-1}^2, \tag{15}$$

along with

$$\overline{\sigma}_k^2 = \xi_k^2 - \overline{\mu}_k^2. \tag{16}$$

6.4 Multidimensional Variance

The moving variance method can be adapted to deal with multidimensional numerical features, i.e., with a plurality of numerical features simultaneously. For network traffic data, these features may correspond to different nodes or links in a network or to different values of the underlying symbolic packet features such as, e.g., the IP addresses of port numbers. Under the assumption that the chosen numerical features

are roughly statistically independent, the proposed aggregated statistical anomaly detection criterion is then the sum of relative squared differences of variances associated with individual numerical features. More generally, the sum can be weighted.

If the numerical features are strongly correlated, then instead of considering the variances only, it is more appropriate to take into account the whole covariance matrices associated with multidimensional data. Accordingly, let $\mathbf{C}_j$ and $\acute{\mathbf{C}}_{j-1}$ denote the estimates of the covariance matrices associated with multidimensional data on the segments of samples $X(j)$ and $\acute{X}(j-1)$ to be compared with each other, respectively. Recall that a generic entry of the covariance matrix of an ordered set of random variables is the covariance between two random variables, where the covariance between a and b is defined as $cov(a,b) = \overline{ab} - \overline{a}\,\overline{b}$. Instead of the sum of relative squared differences of variances, the criterion proposed is then

$$\delta_j = tr(\mathbf{C}_j - \acute{\mathbf{C}}_{j-1})(\acute{\mathbf{C}}_{j-1}^{-1} - \mathbf{C}_j^{-1}), \tag{17}$$

where $tr(\cdot)$ denotes the usual trace operator, i.e., the sum of elements on the main diagonal of a quadratic matrix. If the features are not correlated, then (17) reduces to the sum of relative squared differences of variances, as desired. For correlated data, (17) may be more effective, but is more complex to compute. The criterion (17) is derived from the symmetrized Kullback-Leibler divergence between two multidimensional Gaussian probability distributions, associated with $X(j)$ and $\acute{X}(j-1)$, by discarding the part involving the mean values.

It is interesting that the two EWMA techniques described in Section 6.2 can be adapted to deal with covariance matrices and (17). More precisely, in analogy with the variance, they should be applied to the equivalent expressions for the covariance $\overline{(a-\overline{a})(b-\overline{b})}$ and $\overline{ab} - \overline{a}\,\overline{b}$, respectively, and comprise the EWMA recursions for the involved mean values.

Note that another option would be to treat multidimensional numerical data as vectors and use as a dispersion measure the mean squared Euclidean distance between the data vectors and the mean data vector. However, this generalization of variance does not appear to be sufficiently sensitive to changes in individual features.

7 Moving Linear Least Square Error Method for Numerical Features

For numerical features, the second dispersion measure proposed is the least mean squared error resulting from a linear (more precisely, affine) approximation of data. This error, which is here called the linear least square error (LLSE), together with the optimum affine approximation minimizing the error, can be obtained by the standard linear regression technique. Note that the variance can be regarded as the least mean squared error resulting from a constant approximation of data, where the optimum constant minimizing this error is the mean value of data. The LLSE is less sensitive to changes in nonstationary or correlated normal data than the variance, as it tends to remove linear trends in time which may occur in real data.

Let $\epsilon^2(j)$ and $\acute{\epsilon}^2(j-1)$ denote the respective LLSE estimates associated with the segments of samples $X(j)$ and $\acute{X}(j-1)$ to be compared with each other at the j-th discrete time $t = m_j$. The relative squared difference of the LLSEs is then defined as

$$\delta_j = \frac{(\epsilon^2(j) - \acute{\epsilon}^2(j-1))^2}{\epsilon^2(j)\acute{\epsilon}^2(j-1)}. \tag{18}$$

The LLSE estimates depend on the timings of individual samples which may be regular or irregular. Let t_i denote the time associated with the sample x_i, for each $i \geq 1$. For the sliding window techniques, let $\mu_t(j) = \left(\sum_{i=m_j-n_j+1}^{m_j} t_i\right)/n_j$ and

$$\sigma_t^2(j) = \frac{1}{n_j} \sum_{i=m_j-n_j+1}^{m_j} (t_i - \mu_t(j))^2 \tag{19}$$

denote the mean value and the variance of the sample timings on the segment $X(j)$, respectively. Further, let

$$cov_{x,t}(j) = \left(\frac{1}{n_j} \sum_{i=m_j-n_j+1}^{m_j} x_i t_i\right) - \mu(j)\mu_t(j) \tag{20}$$

denote the covariance between x and t on $X(j)$. According to the standard linear regression technique, we then obtain

$$\epsilon^2(j) = \sigma^2(j) - \frac{cov_{x,t}(j)^2}{\sigma_t^2(j)}, \tag{21}$$

which, in the case of regular data sampling, reduces to

$$\epsilon^2(j) = \sigma^2(j) - \frac{\left(\frac{1}{n_j}\sum_{i=1}^{n_j} i x_{i+m_j-n_j} - \mu(j)\frac{n_j+1}{2}\right)^2}{(n_j^2-1)/12}. \tag{22}$$

Analogous expressions hold for $\acute{\epsilon}^2(j-1)$, associated with $\acute{X}(j-1)$. Iterative update expressions corre-

sponding to (22) are given in the Appendix.

In view of (21), it follows that the LLSE dispersion measure will be close to the variance if the covariance between x and t on $X(j)$ is relatively small. In turn, it can be shown that this will hold if the data samples are statistically independent or, more generally, uncorrelated (i.e., if the covariance between different data samples is relatively close to zero) and stationary. In the opposite case, if the data samples are nonstationary (e.g., if the probability distribution of samples changes in time) or correlated, then the LLSE may be considerably smaller than the variance, as the linear trends in data are removed. Therefore, the LLSE is more robust as a dispersion measure than the variance, i.e., less sensitive to changes in nonstationary or correlated normal data, which may be desirable. In particular, normal data may become nonstationary if ΔT is relatively large and correlated if ΔT is relatively small.

It is interesting that the LLSE can also be estimated by the adapted EWMA techniques. To this end, let us put (21) in the following general, self-explanatory form

$$\begin{aligned}\epsilon^2(x) &= \sigma^2(x) - \frac{cov(x,t)^2}{\sigma^2(t)} \\ &= \sigma^2(x) - \frac{(\overline{xt} - \overline{x}\overline{t})^2}{\sigma^2(t)}.\end{aligned} \tag{23}$$

Now, expressions for $\bar{t}$ and $\sigma^2(t)$ are deterministic and depend on the data sampling used. The EWMA recursion can be applied to the mean value $\overline{x}$, as described above, and to the mean value $\overline{xt}$. More precisely, an EWMA estimate η_k of the mean value $\overline{xt}$ on a generic segment of samples $X_1^k = (x_i)_{i=1}^k$ taken at times $(t_i)_{i=1}^k$ can be iteratively computed as

$$\eta_k = \beta(x_k t_k) + (1-\beta)\eta_{k-1} \tag{24}$$

with the initial value $\eta_1 = x_1 t_1$. Then, by applying the two EWMA techniques from Section 6.2 for estimating the variance we respectively obtain two corresponding EWMA techniques for estimating the LLSE. Depending on the constants used, one may thus obtain even more robust dispersion measures.

8 Moving Concentration Method for Symbolic Features

A symbolic feature x takes values in a set in which a distance or closeness between two elements is not defined or is not considered to be relevant. Typically, such a set is discrete, i.e., finite or countably infinite, and is here denoted as $\mathcal{A} = \{a_k | 1 \le k \le m\}$ and called an alphabet, where the number of elements m is finite or possibly infinite. Multidimensional symbolic features can be treated in essentially the same way and, if m is very large, then the number of elements can be effectively reduced by grouping the discrete values, e.g., by applying a hash function. Any numerical feature can be treated as symbolic, possibly after quantization, by disregarding the Euclidean metric.

A dispersion or concentration measure associated with a multiset of observed symbolic values is a measure of how these values are dispersed or concentrated in a given alphabet, respectively. A dispersion (resp. concentration) measure is a real-valued function that achieves its minimum (resp. maximum) value if all the data values are identical, generally increases (resp. decreases) as the data values become dispersed among a larger subset of values, and achieves its maximum (resp. minimum) value if the data values are uniformly distributed over the whole alphabet. Accordingly, it is natural to associate a dispersion or concentration measure with relative frequencies of observed symbolic values. If these frequencies are interpreted as estimated probabilities, then a sample estimate of any entropy, as a measure of statistical uncertainty, can be taken as a dispersion measure and, in particular, the well-known Shannon entropy.

A relative difference of consecutive estimates of entropy should then be adapted to the chosen entropy and possibly defined according to the statistical properties of entropy estimates derived in [20]. More precisely, a general criterion to be respected in this regard is that, if the samples are drawn independently according to the same probability distribution, then the relative difference should generally be a small random variable that should decrease as the (effective) number of samples per segment increases and should be roughly independent of the underlying probability distribution.

For the quadratic entropy or quadratic concentration measure to be defined below, three relative squared differences are proposed in view of [20]. In a concrete application, the one whose variation over normal data is close to minimal could preferably be chosen. Let $C(j)$ and $\acute{C}(j-1)$ denote the respective estimates of the quadratic concentration measure associated with the segments of samples $X(j)$ and $\acute{X}(j-1)$ to be compared with each other at the j-th discrete time $t = m_j$. Note that the values of the quadratic concentration measure belong to $(0, 1]$. The relative squared differences proposed are then

$$\delta_j = \frac{(C(j) - \acute{C}(j-1))^2}{C(j)\acute{C}(j-1)} \tag{25}$$

$$\delta_j = \frac{(C(j) - \acute{C}(j-1))^2}{(1 - C(j))(1 - \acute{C}(j-1))} \tag{26}$$

$$\delta_j = \frac{(C(j) - \acute{C}(j-1))^2}{\sqrt{C(j)\acute{C}(j-1)(1 - C(j))(1 - \acute{C}(j-1))}}. \tag{27}$$

In addition, some other concentration measures are also proposed in the sequel.

8.1 Sliding Window Techniques

For a segment of samples $X(j)$, let $F_k(j)$ denote the number of times a value a_k is achieved, i.e., the absolute frequency of this value and let

$$f_k(j) = \frac{F_k(j)}{n_j} \tag{28}$$

denote the relative frequency of the value a_k on $X(j)$. The relative frequencies constitute the observed probability distribution of the considered symbolic feature x on $X(j)$. Note that the absolute frequencies can as well be computed if the observed samples do not correspond to individual symbolic values, but to their absolute frequencies on elementary time intervals used for data monitoring (e.g., ΔT).

The quadratic concentration measure is then defined as

$$C(j) = \sum_{k=1}^{m} f_k(j)^2. \tag{29}$$

The related concentration measure

$$m \sum_{k=1}^{m} f_k(j)^2 - 1 = \frac{\sum_{k=1}^{m} (f_k(j) - 1/m)^2}{1/m} \tag{30}$$

is the chi-square statistic of the observed probability distribution with respect to the uniform probability distribution. This statistic follows the chi-square probability distribution if the observed values are drawn from the uniform probability distribution, provided that the number of samples is sufficiently large. The corresponding quadratic dispersion measure

$$D(j) = \sum_{k=1}^{m} f_k(j)(1 - f_k(j)) = 1 - C(j) \tag{31}$$

is the quadratic entropy [21] of the observed probability distribution.

Consequently, the essence of the proposed moving quadratic concentration method is thus to compare the observed probability distributions for two consecutive moving windows by comparing the corresponding chi-square statistics with respect to the uniform probability distribution, instead of comparing them directly by the (two-sample) chi-square statistic. Namely, the direct comparison is in general overly sensitive for normal data, where the observed probability distributions may change rapidly and considerably.

It follows that

$$\frac{1}{m} \leq C(j) \leq 1. \tag{32}$$

The maximum value $C(j) = 1$ is achieved if and only if the observed probability distribution is maximally concentrated, i.e., there exists exactly one relative frequency equal to 1 and all the others are equal to 0. The minimum value $C(j) = 1/m$ is achieved if and only if the observed probability distribution is uniform over all m values, i.e., $f_k(j) = 1/m$, for all $1 \leq k \leq m$. Note that the Shannon entropy $-\sum_{k=1}^{m} f_k(j) \log f_k(j)$ possesses analogous properties, but is much more sensitive to changes in small relative frequencies than the quadratic entropy, and this may not be desirable.

The quadratic concentration measure is particularly interesting if the number of samples n_j is larger than the total number of achievable values m. However, if $n_j < m$, then some or many discrete values cannot effectively appear and it may be advantageous to use alternative measures. One option would be to look at a subset of m' highest relative frequencies only, where $m' < m$. More precisely, let $f'_{k,m'}(j)$, $1 \leq k \leq m'$, denote the normalized m' highest relative frequencies on the segment $X(j)$. Then, given a parameter $m' \leq m$, the proposed quadratic concentration measure is defined as

$$C'(j) = \sum_{k=1}^{m'} f'_{k,m'}(j)^2. \tag{33}$$

It follows that

$$\frac{1}{m'} \leq C'(j) \leq 1, \tag{34}$$

where the maximum value $C'(j) = 1$ is achieved if and only if the observed probability distribution is maximally concentrated and the minimum value $C'(j) = 1/m'$ is achieved if and only if the observed probability distribution is uniform over a subset of m' values.

Another, less sensitive option, which is particularly interesting if n_j is much smaller than m, would be to define a concentration measure as the total number of repetitions among all n_j samples, i.e., as

$$C''(j) = n_j - m_{\text{eff}}(j), \tag{35}$$

where $m_{\text{eff}}(j)$ is the total number of discrete values that effectively appear in $X(j)$, i.e., the total number of nonzero relative frequencies. It follows that

$$0 \le C''(j) \le n_j - 1, \tag{36}$$

where the maximum value $C''(j) = n_j - 1$ is achieved if and only if the observed probability distribution is maximally concentrated, i.e., $m_{\text{eff}}(j) = 1$ and the minimum value $C''(j) = 0$ is achieved if and only if there are no repetitions, i.e., $m_{\text{eff}}(j) = m$, for which it is necessary that $n_j \le m$. Note that if the samples are drawn from the uniform probability distribution, then the expected number of repetitions is approximately $n_j - m(1-e^{-n_j/m})$ and, in particular, if $n_j \approx \sqrt{m}$, then it is approximately equal to $n_j^2/(2m)$. The relative squared difference (25) can be used for comparison.

Unlike the quadratic concentration measures, the repetition concentration measure is sensitive to changes in the number of samples in the two segments to be compared with each other. Note that for the sliding window pair technique, this number necessarily changes as one segment is a subset of the other. One can then perform a normalization of (35) by dividing it by an appropriate normalization factor, e.g., by $n_j^2/(2m)$.

For the sliding window pair technique, the concentration measures $\hat{C}(j-1)$, $\hat{C}'(j-1)$, and $\hat{C}''(j-1)$, in terms of the frequencies $\hat{F}_k(j-1)$ and $\hat{f}_k(j-1)$ on a sample segment $\hat{X}(j-1)$, are defined analogously. Iterative update expressions for the concentration measures are given in the Appendix.

8.2 EWMA Techniques

It is interesting to see if the EWMA technique can be adapted to provide iterative estimates of the quadratic concentration measures (29) and (33) introduced in Section 8.1. A novel observation is that this can be achieved by applying the EWMA recursion to appropriately defined elementary relative frequencies and, then, by computing the concentration measures by using the EWMA estimates of relative frequencies. This holds for any concentration or dispersion measure based on relative frequencies.

Firstly, assume that the exponential weights are associated with individual samples in the sample sequence $X = (x_i)_{i=1}^{\infty}$. Let $\boldsymbol{f}(t) = (f_k(t))_{k=1}^m$ denote a vector of estimated relative frequencies on the segment $X_1^t = (x_i)_{i=1}^t$, to be computed iteratively for any discrete time $t \ge 1$. Let $\boldsymbol{\lambda}(t) = (\lambda_k(t))_{k=1}^m$ denote the value-indicator vector at time t defined by $\lambda_k(t) = [x_t = a_k]$, which contains exactly one component equal to 1, i.e., the one corresponding to the index of the discrete value taken by x_t, and all the others equal to 0.

The estimated relative frequencies are then computed iteratively as

$$\boldsymbol{f}(t) = \alpha\boldsymbol{\lambda}(t) + (1-\alpha)\boldsymbol{f}(t-1) \tag{37}$$

with the initial value $\boldsymbol{f}(1) = \boldsymbol{\lambda}(1)$ or, in scalar notation, as

$$f_k(t) = \alpha\lambda_k(t) + (1-\alpha)f_k(t-1), \tag{38}$$

for each $1 \le k \le m$. The explicit solution is then given as

$$\boldsymbol{f}(t) = (1-\alpha)^{t-1}\boldsymbol{\lambda}(1) + \alpha\sum_{i=2}^{t}(1-\alpha)^{t-i}\boldsymbol{\lambda}(i) \tag{39}$$

or, in scalar notation, as

$$f_k(t) = (1-\alpha)^{t-1}\lambda_k(1) + \alpha\sum_{i=2}^{t}(1-\alpha)^{t-i}\lambda_k(i). \tag{40}$$

The constant α should be sufficiently small so that the equivalent number of samples $2/\alpha - 1$ is sufficiently large for obtaining meaningful estimates of relative frequencies.

Alternatively, the exponential weights can be associated not with individual samples, but with groups of successive samples, in which case the constant α is not required to be very small. The discrete times are then associated with these groups of samples instead of individual samples. In particular, groups of samples may correspond to elementary time intervals of equal lengths (e.g., ΔT). For individual samples, the value-indicator vector for an individual sample can be regarded as the vector of elementary relative frequencies corresponding to this sample, which are hence equal to 0 or 1. For groups of samples, at a discrete time t corresponding to the t-th group of samples, we can thus define $\overline{\boldsymbol{\lambda}}(t) = (\overline{\lambda}_k(t))_{k=1}^m$ as the vector of elementary relative frequencies of values occurring in this group, which is equal to the arithmetic mean of the value-indicator vectors corresponding to individual samples in this group. The EWMA recursion for the estimated relative frequencies then becomes

$$\boldsymbol{f}(t) = \alpha\overline{\boldsymbol{\lambda}}(t) + (1-\alpha)\boldsymbol{f}(t-1) \tag{41}$$

with the initial value $\boldsymbol{f}(1) = \overline{\boldsymbol{\lambda}}(1)$.

The two described EWMA techniques can be called the sample-based and the interval-based EWMA

techniques for relative frequencies. For both of them, the quadratic concentration measures $C(j)$ and $C'(j)$, at a discrete time $t = m_j$, are then computed by applying (29) and (33) to $\boldsymbol{f}(m_j)$, respectively.

8.3 Average Conditional Concentration Measures

Let a symbolic feature x be associated with another symbolic feature s, i.e., let us consider a two-dimensional symbolic feature (x, s). For example, for network traffic data, the two symbolic features may relate to the source and destination variables such as the IP addresses, port numbers, email addresses, HTTP URIs, and SIP URIs. In this case, one can define the quadratic or repetition concentration measures for the joint variable (x, s) and for individual variables x and s, in the same way as above.

However, it may be advantageous to define the conditional concentration measure of x, conditioned on s, and then averaged over s. Of course, x and s may switch places. To this end, we need to compute the conditional relative frequencies

$$f_{k_1|k_2}(j) = \frac{f_{k_1,k_2}(j)}{f_{k_2}(j)} \tag{42}$$

for the values a_{k_2} of s that effectively appear, i.e., such that $f_{k_2}(j) > 0$, where $f_{k_2}(j) = \sum_{k_1=1}^{m} f_{k_1,k_2}(j)$. For the sliding window technique, the average quadratic concentration measure of x conditioned on s, on the segment $X(j)$, is then given as

$$\overline{C}(j) = \sum_{k_2=1}^{m} f_{k_2}(j) \sum_{k_1=1}^{m} f_{k_1|k_2}(j)^2 \tag{43}$$

$$= \sum_{k_1,k_2=1}^{m} \frac{f_{k_1,k_2}(j)^2}{f_{k_2}(j)}, \tag{44}$$

and $\hat{\overline{C}}(j-1)$ is computed similarly. Analogous expressions hold for the average conditional quadratic measure based on (33) instead of (29). The average conditional repetition concentration measure can be defined similarly.

The EWMA techniques proposed in Section 8.2 can be adapted to deal with the average conditional quadratic concentration measures. One option is to apply the EWMA recursions to the relative frequencies $f_{k_1,k_2}(t)$ and $f_{k_2}(t)$, where the constant used for the joint relative frequency should roughly be m times smaller, because of the m times larger number of two-dimensional values. Another, more direct option is to apply the EWMA recursions to $f_{k_2}(t)$ and the individual conditional relative frequencies $f_{k_1|k_2}(t)$, where the constants can be the same or similar. For each value of k_2, $f_{k_1|k_2}(t)$ is then iteratively updated irregularly in time, whenever a particular value of k_2 appears. The corresponding average conditional quadratic concentration measures are computed by (44) and (43), respectively. For each option, the EWMA recursions can be sample based or interval based, as described in Section 8.2.

9 Network Traffic Applications

For network traffic data, as discussed in Section 3, the main numerical features are extracted from layers 3 and 4 of IP packet headers in elementary time intervals of length ΔT and include the packet rate R_{packet}, the byte rate R_{byte}, and the average packet size P_{size}. The main symbolic features extracted from layers 3 and 4 of individual IP packet headers include the source and destination IP addresses and port numbers and the transport protocol used, whereas the symbolic features extracted from other layers such as the application layer may include the source and destination HTTP URIs, SIP URIs, and email addresses, as well as the type of SIP packets transmitted in a VoIP network. These features can be traced in time at a chosen network node or a set of nodes (e.g., routers or points of presence), possibly distinguishing the direction of packets along communication links. Typically, ΔT can range from being relatively small (on the order of seconds or less) to relatively large (on the other of minutes, e.g., 5 min).

Numerical features can be aggregated or separated according to various classes of symbolic features, thus resulting in various combined or multidimensional features containing both numerical and symbolic data. Numerical features can also be regarded as symbolic, possibly after quantization. On the other hand, one may only consider the symbolic features, discarding the numerical ones. Note that, in a given elementary time interval, N_{packet} associated with different values of a symbolic feature (e.g., destination port number) in fact represents, after normalization, the elementary relative frequency distribution of this feature in this time interval. It can as well be used for the computation of relative frequencies and concentration measures as explained in Section 8, either by sliding window techniques or the interval-based EWMA technique, with the time resolution corresponding to ΔT instead of individual samples.

Let dPN and sPN stand for destination and source port numbers and $dIPA$ and $sIPA$ for destination and source IP addresses, respectively. Some examples of numerical features to be considered include overall numerical features R_{packet}, R_{byte}, and P_{size} aggregated over all port numbers and IP addresses, condi-

tional numerical features $R_{packet}|dPN$, $R_{byte}|dPN$, and $P_{size}|dPN$ conditioned on dPN and aggregated over sPN and IP addresses, $R_{packet}|dIPA$, $R_{byte}|dIPA$, and $P_{size}|dIPA$ conditioned on $dIPA$ and aggregated over port numbers and $sIPA$, and $R_{packet}|sIPA$, $R_{byte}|sIPA$, and $P_{size}|sIPA$ conditioned on $sIPA$ and aggregated over port numbers and $dIPA$. Other examples are obtained analogously. If in a conditional numerical feature the value of the conditioning variable is not fixed, then we obtain a combined feature for which one may compute average conditional dispersion measures. If ΔT is relatively small or relatively large, then it may be more appropriate to use the LLSE rather than the variance as a dispersion measure.

Some examples of symbolic features include dPN aggregated over sPN and IP addresses, sPN aggregated over dPN and IP addresses, $dIPA$ aggregated over port numbers and $sIPA$, and $sIPA$ aggregated over port numbers and $dIPA$. Conditional symbolic features include $dPN|sPN$, $sPN|dPN$, $dIPA|sIPA$, and $sIPA|dIPA$, for which, if the value of the conditioning variable is not fixed, one may then compute average conditional concentration measures. Other examples with these features are obtained analogously (e.g., $(dIPA, sIPA)|dPN$ and $dIPA|dPN$). In addition, instead of port numbers and IP addresses, one can analogously consider other symbolic features like email addresses, HTTP URIs, or SIP URIs.

The proposed moving dispersion or moving concentration method is potentially useful for detecting network failures and broad classes of network attacks such as (D)DoS attacks, scanning or probing attacks (e.g., port-scanning attacks), SPAM and SPIT attacks, as well as worm or virus outbreaks and other massive malicious software attacks. The underlying expectation is that the relative change in time of the dispersion or concentration measure between two consecutive moving windows may be much smaller in normal traffic than when there is a transition of normal traffic into anomalous. The rationale for this expectation which is not intended to be exhaustive is emphasized in the sequel.

Firstly, in a message or request flooding (D)DoS attack, the anomalous traffic may consist of a repeated transmission of essentially the same or similar packets (payload included), if the same piece of data or code is distributed over the network. For example, in a SIP network, a particular type of SIP messages/packets (e.g., INVITE, RE-INVITE, BYE, or REGISTER) may become much more frequent than the others in case of such an attack. This may also happen in case of network failures (e.g., in case of failure of a REGISTER server in a SIP network, the network may become flooded by REGISTER messages). Then the variance or LLSE of P_{size} for SIP packets, possibly conditioned on $dIPA$ or the destination SIP URI, is expected to decrease considerably and, similarly, the quadratic concentration measure of (possibly quantized) P_{size} is expected to increase.

Secondly, in a DDoS attack, the source IP addresses tend to become randomized, especially near the target(s). In particular, this occurs if the source IP addresses are randomly spoofed. As a consequence, the quadratic or repetition concentration measure of the symbolic feature $sIPA$ tends to decrease significantly. Moreover, if the number of targeted network nodes is small, like in a reflection DDoS attack in a SIP network, then the quadratic concentration measure of the symbolic feature $dIPA$ tends to increase, especially near the target(s). Moreover, the average conditional quadratic concentration measure of the conditional symbolic feature $sIPA|dIPA$ tends to decrease with even higher sensitivity. A similar situation occurs in a DDoS attack with randomized source email addresses or source SIP URIs.

Thirdly, in a port scanning attack, which usually precedes a DDoS attack, the destination port numbers in the corresponding network packets may become randomly dispersed. The quadratic concentration measure of the symbolic feature dPN and the average conditional quadratic concentration measure of the symbolic feature $dPN|sPN$ then tend to decrease significantly.

Fourthly, in a massive malicious software attack targeting random or random-like destination IP addresses, the quadratic concentration measure of the symbolic feature $dIPA$ and the average conditional quadratic concentration measure of the conditional symbolic feature $dIPA|sIPA$ tend to decrease significantly. If such an attack targets specific destination port numbers, then the quadratic concentration measure of dPN tends to increase. In addition, the variance or LLSE of P_{size} or R_{byte}, possibly conditioned on dPN, may then change considerably. If the source IP addresses are randomly spoofed, then the joint feature $(dIPA, sIPA)$ can be used similarly.

Fifthly, in a SPAM or SPIT attack using random destination email addresses or SIP URIs, the quadratic concentration measure of these addresses or URIs tends to decrease considerably, respectively. The same is the case with the average conditional quadratic concentration measure of these addresses or URIs, when conditioned on source email addresses or SIP URIs, respectively, and the sensitivity to such attacks is expected to be even higher.

For network traffic data, one can distinguish between two directions of traffic along a communication link, e.g., inbound and outbound with respect to a given network. Instead of looking for sudden relative changes in time of a chosen dispersion measure associated with a chosen data feature, one may as well apply another paradigm, namely, that of looking for points in time when the relative difference between the two values of the chosen dispersion measure corresponding to two directions of traffic is large. Namely, if in a normal traffic there is a symmetry between the two directions so that the considered relative differences are likely to be small, then in a case of an attack such a symmetry may be broken, due to functional disruptions or protective countermeasures taking place, which in turn may result in larger values of the relative differences.

Finally, in addition to dispersion or concentration measures, one may also consider standard volume-based techniques aiming at detecting sudden increases in the mean values of R_{packet} and R_{byte}, which accompany the flooding attacks and then apply combined decision criteria for anomaly detection. These combined criteria should also reflect a plurality of dispersion and concentration measures possibly used simultaneously for statistical anomaly detection.

The volume-based techniques are not sufficient by themselves since a significant traffic volume increase may also be caused by normal traffic such as flash crowds and since the flooding attacks need to be detected earlier, before the traffic volume becomes excessively large. For example, in case of DDoS attacks, the traffic volume becomes high near the target, but is low near the distributed sources of the attack. Since the techniques based on dispersion or concentration measures are less sensitive to traffic load increases and more sensitive to other traffic anomalies, namely, those featuring a relative change in the dispersion of data, they will issue less alerts in the case of normal traffic volume increases and may be able to detect anomalous traffic even when the traffic volume is relatively low, respectively.

10 Conclusions

The moving dispersion method is proposed as a novel general framework for statistical anomaly detection in intrusion detection systems. It essentially consists in computing a chosen dispersion or concentration measure of numerical or symbolic data at successive discrete times and finding the times when its values change significantly, with respect to appropriately defined relative differences. For stationary data, the relative differences should be small and roughly insensitive to data, so that the corresponding thresholds, possibly dynamic, should reflect only the nonstationary nature of data. The proposed dispersion or concentration measures include the variance and the least square error resulting from the linear regression technique (LLSE), for numerical data, and the quadratic and repetition concentration measures, for symbolic data. They are proposed to be iteratively estimated in time on three types of moving windows including the standard sliding window technique, a novel sliding window pair technique, and a number of new EWMA techniques adapted to the dispersion or concentration measures considered.

While its origins can be found in previous work, e.g., dealing with the variance of byte or packet rates in network traffic and the Shannon entropies or compression ratios for IP addresses and port numbers, apart from the relative change criterion, the new method and the corresponding techniques contain many new elements and generalizations and arguably provide a number of advantages in terms of reduced costs and potentially increased effectiveness. This remains to be experimentally tested on simulated and real data and is out of the scope of this paper. The experiments to be conducted may also relate to the scenario where the attack parameters are adapted to the statistical anomaly detection techniques applied.

For numerical data, new elements include the relative squared difference of variances and its generalization to multidimensional data in terms of covariance matrices, the LLSE as a dispersion measure for nonstationary or correlated normal data, the average conditional variance as a dispersion measure of numerical data conditioned on symbolic data, and the EWMA techniques for the estimation of the proposed dispersion measures. For symbolic data, new elements include the quadratic concentration measure and the corresponding relative differences, the repetition concentration measure, the average conditional quadratic and repetition concentration measures of symbolic data conditioned on symbolic data, and the EWMA techniques for the estimation of the proposed concentration measures.

The average conditional dispersion and concentration measures have a higher sensitivity to massive network attacks that follow the connection strategies of the types many-to-few or few-to-many. The EWMA techniques are better suited to nonstationary normal data and enable a significant reduction of the amount of data needed to be memorized, in comparison with the sliding window techniques.

The proposed method is not computationally demanding and enables in-line statistical anomaly detection in real time, even in high-speed and high-

volume communications networks. In principle, it does not require prior complex training on historical data for deriving the underlying statistical models, but only for determining the corresponding static or dynamic thresholds.

Appendix A: Iterative Update Expressions

A1 Single Sliding Window and Variance

If two consecutive segments $X(j)$ and $X(j-1)$ have a lot of samples in common, then $\sigma^2(j)$ can be computed by updating $\sigma^2(j-1)$ on the basis of (2). Let $\Delta_j = \max(m_j - m_{j-1}, m_j - m_{j-1} - n_j + n_{j-1})$, where $m_j - m_{j-1}$ is the number of samples in $X(j)$ not in $X(j-1)$ and $m_j - m_{j-1} - n_j + n_{j-1}$ is the number of samples in $X(j-1)$ not in $X(j)$. Let $S_1(j) = \sum_{i=m_j-n_j+1}^{m_j} x_i$ and $S_2(j) = \sum_{i=m_j-n_j+1}^{m_j}(x_i - \mu_(j))^2$. Note that $\mu(j) = S_1(j)/n_j$ and $\sigma_j^2 = S_2(j)/n_j$. Further, define $\mu'(j-1) = S_1(j-1)/n_j$.

Initially, first compute $S_1(1)$, $\mu(1)$, $S_2(1)$, and $\sigma^2(1)$. Then, for any $j \geq 2$, iteratively update $\mu(j-1)$ into $\mu(j)$ and $\sigma^2(j-1)$ into $\sigma^2(j)$, by using the following update expressions for the sums $S_1(j)$ and $S_2(j)$

$$S_1(j) = S_1(j-1) + \sum_{i=m_j-\Delta_j+1}^{m_j} (x'_i - x'_{i-n_j}) \qquad (45)$$

$$S_2(j) = S_2(j-1) + \mu(j-1)\mu'(j-1)(n_{j-1} - n_j) + \sum_{i=m_j-\Delta_j+1}^{m_j} (x'_i - x'_{i-n_j})(x'_i - \mu(j) + x'_{i-n_j} - \mu'(j-1)) \qquad (46)$$

where $x'_i = x_i$ if x_i is not contained in $X(j-1)$ (i.e., if $i \geq m_{j-1}+1$), $x'_{i-n_j} = x_{i-n_j}$ if x_{i-n_j} is contained in $X(j-1)$ (i.e., if $i \geq m_{j-1} - n_{j-1} + n_j + 1$), and $x'_i = 0$ or $x'_{i-n_j} = 0$ otherwise.

If the numbers of samples in $X(j)$ and $X(j-1)$ are equal, i.e., $n_j = n_{j-1}$, then $\Delta_j = m_j - m_{j-1}$ and the update expressions simplify into

$$S_1(j) = S_1(j-1) + \sum_{i=m_{j-1}+1}^{m_j} (x_i - x_{i-n_j}) \qquad (47)$$

$$S_2(j) = S_2(j-1) + \sum_{i=m_{j-1}+1}^{m_j} (x_i - x_{i-n_j})(x_i - \mu(j) + x_{i-n_j} - \mu(j-1)). \qquad (48)$$

A2 Sliding Window Pair and Variance

Since the shortened segment $\hat{X}(j-1)$ is contained in $X(j)$, it is not efficient to compute $\sigma^2(j)$ and $\hat{\sigma}^2(j-1)$ by (2) and (3), respectively, for each j. Instead, it is efficient to compute iteratively $\hat{\sigma}^2(j-1)$ by using the update expressions from Section 10, applied to the shortened sliding window, and then, for each j, to update $\hat{\mu}(j-1)$ into $\mu(j)$ and $\hat{\sigma}^2(j-1)$ into $\sigma^2(j)$ by using

$$\mu(j) = \hat{\mu}(j-1) + \frac{1}{n_j} \sum_{i=m_{j-1}+1}^{m_j} (x_i - \hat{\mu}(j-1)) \qquad (49)$$

$$S_2(j) = \hat{S}_2(j-1) + \sum_{i=m_{j-1}+1}^{m_j} (x_i - \hat{\mu}(j-1))(x_i - \mu(j)), \qquad (50)$$

where $\hat{S}_2(j-1) = \sum_{i=m_j-n_j+1}^{m_{j-1}} (x_i - \hat{\mu}(j-1))^2$ and $\hat{\sigma}^2(j-1) = \hat{S}_2(j-1)/\hat{n}_{j-1}$. Note that (49) and (50) generalize the well-known update expressions [22], which hold for $m_j = m_{j-1} + 1$.

A3 Sliding Windows and LLSE

An iterative update expression for $\epsilon^2(j)$ given by (22) can be obtained by using the iterative update expressions for $\mu(j)$ and $\sigma^2(j)$ from Section 10 and an update expression for the sum $S'_1(j) = \sum_{i=1}^{n_j} i x_{i+m_j-n_j}$

$$S'_1(j) = S'_1(j-1) + \sum_{i=n_{j-1}+1}^{m_j-m_{j-1}+n_{j-1}} i x_{i+m_{j-1}-n_{j-1}} - \sum_{i=1}^{m_j-m_{j-1}-\Delta n_j} i x_{i+m_{j-1}-n_{j-1}} - (m_j - m_{j-1} - n_j + n_{j-1}) S_1(j), \qquad (51)$$

where $\Delta n_j = n_j - n_{j-1}$.

For the sliding window pair technique, one can proceed along similar lines as in Section 10 by using the iterative update expressions for $\mu(j)$ and $\sigma^2(j)$, along with

$$S'_1(j) = \hat{S}'_1(j-1) + \sum_{i=n_j-m_j+m_{j-1}+1}^{n_j} i x_{i+m_j-n_j}, \qquad (52)$$

where $\hat{S}'_1(j-1) = \sum_{i=1}^{n_j-m_j+m_{j-1}} i x_{i+m_j-n_j}$.

A4 Sliding Windows and Concentration Measures

For the sample segments $X(j)$ and $\acute{X}(j-1)$ to be compared with each other, let $F_k^{\text{new}}(j)$ and $\acute{F}_k^{\text{old}}(j-1)$ denote the absolute frequencies of a_k among the samples in $X(j)$ not in $\acute{X}(j-1)$ and the samples in $\acute{X}(j-1)$ not in $X(j)$, respectively. For a single sliding window, if T is an integer multiple of τ, then $F_k^{\text{old}}(j-1)$ need not be recomputed as it is equal to a previously computed and memorized value $F_k^{\text{new}}(j-T/\tau)$. For two sliding windows, $\acute{F}_k^{\text{old}}(j-1) = \hat{F}_k^{\text{old}}(j-1) = 0$. The update expressions for the concentration measures are then based on the update expression for the absolute frequencies

$$F_k(j) = \acute{F}_k(j-1) + F_k^{\text{new}}(j) - \acute{F}_k^{\text{old}}(j-1), \qquad (53)$$

together with $f_k(j) = F_k(j)/n_j$ as well as $m_{\text{eff}}(j) = \sum_{k=1}^{m}[F_k(j) > 0]$.

More precisely, one can then compute

$$C(j) = \acute{C}(j-1) + \sum_{k=1}^{m} (f_k(j) - \acute{f}_k(j-1))(f_k(j) + \acute{f}_k(j-1)) \qquad (54)$$

$$C'(j) = \acute{C}'(j-1) + \sum_{k=1}^{m'} (f'_{k,m'}(j) - \acute{f}'_{k,m'}(j-1))(f'_{k,m'}(j) + \acute{f}'_{k,m'}(j-1)) \qquad (55)$$

$$C''(j) = \acute{C}''(j-1) + (n_j - \acute{n}_{j-1}) - (m_{\text{eff}}(j) - \acute{m}_{\text{eff}}(j-1)), \qquad (56)$$

where $\acute{n}_{j-1}$ is the number of samples in $\acute{X}(j-1)$. For the single sliding window technique, if $n_j = n_{j-1}$, then (54) reduces to

$$C(j) = C(j-1) + \sum_{k=1}^{m} (F_k^{\text{new}}(j) - F_k^{\text{old}}(j-1)) \frac{2F_k(j-1) + F_k^{\text{new}}(j) - F_k^{\text{old}}(j-1)}{n_j^2}, \qquad (57)$$

where the summation is only over the values of k such that $F_k^{\text{new}}(j) \neq F_k^{\text{old}}(j-1)$. The update expressions are effective if the number of discrete values m is large.

Acknowledgements

The author would like to thank Rosalia D'Alessandro for contributing to the concept of variance estimation by EWMA techniques and Paola Petiva for a prior art search and a number of references pointed out.

References

[1] L. Portnoy, E. Eskin, and S. Stolfo. Intrusion Detection with Unlabeled Data Using Clustering. In *Proceedings of the ACM CSS Workshop on Data Mining Applied to Security (DMSA'01)*, Philadelphia, PA, USA, 2001.

[2] S.C. Lee and D.V. Heinbuch. Training a Neural-Network Based Intrusion Detector to Recognize Novel Attacks. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 31(4):294–299, 2001.

[3] A. Lakhina, M. Crovella, and C. Diot. Mining Anomalies Using Traffic Feature Distributions. In *Proceedings of the ACM 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM'05)*, pages 217–228, Philadelphia, Pennsylvania, USA, 2005.

[4] G.L. MacIsaac. Network Bandwidth Anomaly Detector Apparatus and Method for Detecting Network Attacks Using Correlation Function. *Patent Application WO 2004/056063 A1*, 2004.

[5] L. Li and G. Lee. DDoS Detection and Wavelets. *Telecommunication Systems - Modeling, Analysis, Design and Management*, 28(3/4):435–451, 2005.

[6] P. Barford and D. Plonka. Characteristics of Network Traffic Flow Anomalies. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pages 69–73, San Francisco, CA, USA, 2001.

[7] N. Ye, C. Borror, and Y. Zhang. EWMA Techniques for Computer Intrusion Detection Through Anomalous Changes in Event Intensity. *Quality and Reliability Engineering International*, 18:443–451, 2002.

[8] D.M. Dempsey. Dynamic Deviation. *Patent US 6,601,014 B1*, 2003.

[9] A.E. Dudfield and M.A. Poletto. Connection Based Denial of Service Detection. *Patent Application US 2004/0220984 A1*, 2004.

[10] J. Viinikka and H. Debar. Monitoring IDS Background Noise Using EWMA Control Charts and Alert Information. In *Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID'04)*, volume 3224 of *Lecture Notes in Computer Science (LNCS)*, pages 166–187, Sophia Antipolis, France, 2004. Springer.

[11] M. Mandjes, I. Saniee, and A.L. Stolyar. Load Characterization and Anomaly Detection for Voice Over IP Traffic. *IEEE Transactions on*

Neural Networks, 16(5):1019–1026, 2005.

[12] L. Feinstein, D. Schnackenberg, R. Balupari, and D. Kindred. Statistical Approaches to DDoS Attack Detection and Response. In *Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX'03)*, volume 1, pages 303–314, Washington, DC, USA, 2003.

[13] K. Kumar, R.C. Joshi, and K. Singh. A Distributed Approach Using Entropy to Detect DDoS Attacks in ISP Domain. In *Proceedings of the IEEE International Conference on Signal Processing, Communications and Networking (ICSCN'07)*, pages 331–337, Chennai, India, 2007.

[14] A. Wagner and B. Plattner. Entropy Based Worm and Anomaly Detection in Fast IP Networks. In *Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'05)*, pages 172–177, Linköping University, Sweden, 2005.

[15] T. Okabe, T. Kitamura, and T. Shizuno. Statistical Traffic Identification Method Based on Flow-Level Behaviour for Fair VoIP Service. In *Proceedings of the 1st IEEE Workshop on VoIP Management and Security*, pages 33–38, Vancouver, BC, Canada, 2006.

[16] C.D. Jeffries, W.J. Jong, G.W. Randall, and K.V. Vu. Detecting Randomness in Computer Network Traffic. *Patent Application US 2003/0200441 A1*, 2003.

[17] T. Peng, C. Leckie, and K. Ramamohanarao. Proactively Detecting Distributed Denial of Service Attacks Using Source IP Address Monitoring. In *Proceedings of the 3rd International IFIP-TC6 Conference on Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications (Networking 2004)*, volume 3042 of *Lecture Notes in Computer Science (LNCS)*, pages 771–782, Athens, Greece, 2004. Springer.

[18] E. Haraldsson. *DDoS Attack Detection Based on Netflow Logs*. Student thesis, Swiss Federal Institute of Technology, Zurich, 2003.

[19] A. Weisskopf. *Plug-ins for DDoS Attack Detection in Realtime*. Semester thesis, Swiss Federal Institute of Technology, Zurich, 2004.

[20] M.D. Esteban and D. Morales. A Summary on Entropy Statistics. *Kybernetika*, 31(4):337–346, 1995.

[21] J.Dj. Golić. On the Relationship Between the Information Measures and the Bayes Probability of Error. *IEEE Transactions on Information Theory*, 33(5):681–693, 1987.

[22] D.E. Knuth. *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*. Addison-Wesley, 1998.

Jovan Dj. Golić received the BSc, MSc, and PhD degrees in electrical engineering from the School of Electrical Engineering, University of Belgrade, Belgrade, Yugoslavia, in 1979, 1981, and 1985, respectively.
From 1979 to 1993, he worked at the Institute of Applied Mathematics and Electronics, Belgrade, where he was appointed a Department Head in 1986 and a Senior Research Fellow in 1990. In 1987 and 1988, he was a Fulbright Visiting Scientist at the School of Electrical Engineering, Cornell University, Ithaca, NY. He was a part-time Docent with the School of Electrical Engineering, University of Belgrade, from 1988 to 1997, as well as with the Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Yugoslavia, from 1986 to 1997. Since 1985, he has been a part-time Research Associate at the Mathematical Institute, Serbian Academy of Science and Arts, Belgrade. From 1993 to 1997, he was a Research Scientist at the Information Security Research Centre, Queensland University of Technology, Brisbane, Australia. From 1997 to 2001, he worked as an Associate Professor with the School of Electrical Engineering, University of Belgrade. From 2001 to 2003, he was a chief cryptographer at Rome CryptoDesign Center, Gemplus, Italy. In 2003, he joined Telecom Italia Lab in Turin and, in 2005, he moved to Security Innovation, Telecom Italia, Turin, Italy.

Prof. Golić has taught and developed undergraduate and graduate courses in cryptology, information theory, data compression and error control coding, algebra, numerical analysis, and discrete mathematics. His research interests include design and cryptanalysis of stream and block ciphers, secure implementation of cryptosystems, sequences, coding and information theory, discrete mathematics, pattern recognition, optimization, biometrics, and statistical anomaly detection. He is a member of the International Association for Cryptologic Research.