# An Observation-Centric Analysis on the Modeling of Anomaly-based Intrusion Detection

Zonghua Zhang, Hong Shen, Yingpeng Sang
*(Corresponding author: Zonghua Zhang)*

School of Information Science, Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikwa, 923-1292, Japan.
Tel: 81-761-51-1209, Fax: 81-761-51-1149

*(Received Aug. 3, 2005; revised and accepted Sept. 21 and 29, 2005)*

## Abstract

It is generally agreed that two key points always attract special concerns during the modelling of anomaly-based intrusion detection. One is the techniques about discerning two classes with different features, another is the construction/selection of the observed sample of normally occurring patterns for system normality characterization. In this paper, instead of focusing on the design of specific anomaly detection models, we restrict our attention to the analysis of the anomaly detector's operating environments, which facilitates us to insight into anomaly detectors' operational capabilities, including their detection coverage and blind spots, and thus to evaluate them in convincing manners. Taking the similarity with the induction problem as the starting point, we cast anomaly detection in a statistical framework, which gives a formal analysis of anomaly detector's anticipated behavior from a high level.

Some existing problems and possible solutions about the normality characterization for the observable subjects that from hosts and networks are addressed respectively. As case studies, several typical anomaly detectors are analyzed and compared from the prospective of their operating environments, especially those factors causing their special detection coverage or blind spots. Moreover, the evaluation of anomaly detectors are also roughly discussed based on some existing benchmarks. Careful analysis shows that the fundamental understanding of the operating environments (i.e., properties of observable subjects) is the elementary but essential stage in the process of establishing an effective anomaly detection model, which therefore worth insightful exploration, especially when we face the dilemma between anomaly detection performance and the computational cost.

*Keywords: Anomaly detection, computer security, information security, intrusion detection, misuse detection*

## 1 Introduction

Intrusion detection is about discerning any intrusive anomalies that might threaten the security from the normal operations/activities of information systems. Existing intrusion detection techniques fall into two general categories: anomaly detection and misuse detection (or signature-based intrusion detection). Anomaly detection techniques mainly focus on establishing normal activities pattern (set or rule) $\Phi$, and any current activity $\phi$ that deviates from $\Phi$ is treated as intrusion. On the contrary, misuse detection techniques attempt to create a model of attack signatures $\Psi$, when a current signature $\psi$ matches with $\Psi$, it is regarded as an intrusion. However, the defects exist in anomaly detection and misuse detection, false positive ($\psi$ is misclassified to $\Phi$) and false negative (novel attack $\psi \in \Psi$ is ignored) respectively, often cause these techniques to fail. Due to the uncontrollable false alarm rate, most of the existing commercial IDSs prefer misuse detection rather than anomaly detection techniques.

It is well known that two elements are essential to design an effective IDS, namely, modelling of the observable subjects and the techniques of characterizing and analyzing the data model. Specifically, several questions should be considered carefully: *What* observable subjects should be selected for monitor and analyse? *What* attributes should be taken into account to characterize those selected subjects? *What* existing approaches or novel methods can be employed to detect anomalies based on the characterized observations? As we have known, network can be logically classified into two parts, hosts and communication links among the hosts. Accordingly, network traffic data, which capture data packets travelling on the communicate links, and audit data, which record the sequence of events on the hosts can be selected as observable subjects. Those two domains actually can be further exploited for seeking more particular and effective observation, such as command line strings, system call traces, and resource consumption patterns in the host audit data, or the intrinsic

features, traffic features and content features of the network packets. Based on the characterization of the data model, all techniques that are capable of distinguishing illegitimate and normal behaviors worth consideration. Up to now, techniques drawn from statistics [7, 27, 28], data mining [13], pattern recognition [3, 30], machine learning [10, 12] and other research fields have been applied to intrusion detection with limited performance.

As we know, the basic assumption for anomaly detection is that the intrinsic characteristic or regularity of the observable subjects deviate significantly from that of anomalies, therefore, the preprocess and analysis of the operating environment, which is composed of specific observations, is an initial but important stage for the modelling of anomaly detectors. In another intuitive explanation, in order to detect anomalies as accurate as possible, meanwhile suppress false alarm rate as low as possible, characterization of the system normality is definitely essential. However, due to the increasing complexity of modern computer systems and the diverse nature of the network, it is generally agreed that there is no such thing as a typical and perfect "system normality description". A possible way, which is also the trend of current anomaly detection research, is to develop methods for characterizing a given operating environment sufficiently well so that optimal detectors for that environment can be designed. The cost must be paid of such work is to allow the limits of detectors, in terms of expected false alarm rate, to be predicted. Along the line, most of the available anomaly detection techniques employ specific subjects with manageable properties as observation, and modelling the subjects as they needed. Although many attacks can be identified using these models, unperfect description of the normality and the novel legitimate activities make them suffer from uncontrollable false alarm rate. Furthermore, most of existing anomaly detectors pay more attention to the technique itself, rather than the fundamental understanding of the working field, which restricts them to a broader application. Additionally, the evaluation of anomaly detectors is deficient and unconvincing due to the limits of so-called benchmark data set, especially for the researches that have been focusing on a specific method for a particular operating environment, which built based solely on "expert" knowledge.

With the introduced problems in mind, our work aims to explore the fundamental attributes of some observable subjects, and analyze the operating environment of several typical anomaly detectors that drawn from different research fields. In general, our work includes:

- Casting the anomaly-based intrusion detection in a statistical modelling framework, and characterize the system normality in a general way by selecting several specific observable subjects that have been applied to some existing typical anomaly detectors;
- Giving a critical analysis of the operating environments that some anomaly detectors work with (mainly the ordering property and frequency property), as well as their operational capabilities/limits and comparative studies;
- Concluding the current evaluation methodologies, and propose our idea for better measurement metrics based on some critical analysis.

The rest of paper is organized as follows. Section 2 constructs a statistical framework to describe anomaly detectors's behavior from a general viewpoint. In Section 3, we give a general description of the selected observation normality. Section 4 characterizes the operating environment of several typical anomaly detectors, together with the analysis of their operational limits. In Section 5, we propose our scheme for better anomaly detection metrics based on some existing contributions. Finally, we give a general discussion in Section 6.

## 2 A General Statistical Description

A general statistical formulation of the computer misuse detection have been discussed in [7], which is generally regarded as a theoretical framework for the latter development of intrusion detection models. With the similar formulation, while pay more attention to the anomaly detectors' operating environments, i.e., the properties of observable subjects, in this section, we give another statistical description for the anomaly detectors' anticipated behavior from a more general viewpoint. The description is based on the analogy between *anomaly detection* and *induction reference* problem.

First of all, several notations need to be given as following for further analysis:

***Notations:***

$H(t)$: a hidden stochastic process which maps the activities of legitimate users and attackers to a finite space $S$ in terms of discrete time step "$t$"; at time step $t$, if $H(t) = 0$, means legitimate user traces is generated, if $H(t) = 1$, means attacker traces is generated, and it is transparent to the anomaly detectors.

$h(x)$: a hidden stochastic process for generating event $x$.

$O_t$: observation that is captured at time interval $t$, it can represent a single event or a group of events according to the specific detection model, and its generation is governed by the hidden process $H$;

$Set(O_t, w)$: a set of observation $O_i$ ($i$ depends on the specific anomaly detection model) with window $w$ at time step $t$.

$N(t)$: a legitimate stochastic process that is generated at time unit $t$, i.e., $H(t) = 0$;

$n(O_t)$: the probability that the subject to be generated by $N(t)$ at time step $t$ is $O_t$, i.e., $Pr\{O_t|H(t) = 0\}$;

$M(t)$: a malicious stochastic process that is generated at time unit $t$, i.e., $H(t) = 1$;

$m(O_t)$: the probability that generated malicious subject at time step $t$ is $O_t$, i.e., $Pr\{O_t|H(t) = 1\}$;

$\phi_i, 0 \leq i \leq Num$: a pattern (or a probability measure) for legitimate activity that is stored in the normal dataset $\Phi$ with size $Num$;

$\tilde{AD}(\cdot)$: the probabilistic anomaly detector with input $O_t$ or $Set(O_t)$ and output is the probability that input is determined as malicious;

$AD(\cdot)$: the deterministic anomaly detector with input $O_t$ or $Set(O_t)$ and output is the binary determination whether input is malicious.

$\lambda$: *a priori* probability that current observable subject is normal, i.e., $\lambda = Pr\{H(t) = 0\}$, and $\lambda$ is close to 1 due to the fact that the number of malicious process is much smaller than that of normal process.

As we know, the objective of anomaly detectors is to capture any malicious subjects that generated by the hidden stochastic process $H(t)$, and what they depend on is a collection of normality characterization of available subjects. Since $Num$, the size of the samples of the normal patterns $\Phi$ is limited, naturally, the most effective observations (or characterized patterns) are desirable. Generally, two properties of the observable subjects, that is, *ordering property* and *frequency property*, can be taken advantage of to construct the system normality according to the correlation of individual observed events $O_t$. Although some anomaly detectors drawn from machine learning (or specification-based techniques) do not take those two properties as their main concern, our analysis is mainly based on this basic taxonomy.

## 2.1 Frequency-Based Analysis

If $O_t$ is taken independently (here $O_t$ is considered as a unit of events), the available observation can be viewed as an unordered collection of subjects in a particular unit, and the consideration of temporal patterns that the observation may contain is excepted. Helman et al. [7] ever gave a thorough analysis for the statistical foundations of computers audit trail with such property, and in such cases, the probability that current subject $O_t$ is malicious can be determined according to Bayes theorem,

$$\begin{aligned} & Pr\{H(t) = 1|O_t\} \\ = & \frac{Pr_1 \cdot Pr\{H(t) = 1\}}{Pr_1 \cdot Pr\{H(t) = 1\} + Pr_2 \cdot Pr\{H(t) = 0\}} \\ = & \frac{Pr_1 \cdot (1-\lambda)}{Pr_1 \cdot (1-\lambda) + Pr_2 \cdot \lambda} \\ = & \frac{m(O_t) \cdot (1-\lambda)}{m(O_t) \cdot (1-\lambda) + n(O_t) \cdot \lambda} \\ = & \frac{c(O_t)}{c(O_t) + \lambda/(1-\lambda)} \\ Pr_1 = & Pr\{O_t|H(t) = 1\},\ Pr_2 = Pr\{O_t|H(t) = 0\} \end{aligned}$$

where $c(O_t) = m(O_t)/n(O_t)$, and $Pr\{H(t) = 1|O_t\} > \alpha$ iff $c(O_t) > \alpha\lambda/(1-\alpha)(1-\lambda)$. Thus it is easy to find that the performance of anomaly detectors is related directly with the value of $Pr\{H(t) = 1|O_t\}$, and it increases with the value of $c(O_t)$. Based on the equation, a simple anomaly detection model can be defined as:

$$\tilde{AD}(O_t) = c(O_t), \quad AD(O_t) = \begin{cases} 0 & \text{if } \tilde{AD}(O_t) < \alpha \\ 1 & otherwise \end{cases}$$

A series of optimality conditions for the above detection model have been discussed in [7], and as they pointed, due to the lack of prior knowledge about $\lambda$, $m(O_t)$, and $n(O_t)$, it is almost impossible to carry it out into practice. Specifically, a good estimates of $\lambda$ and a thorough understanding of distributions of the processes $N(t)$ and $M(t)$, which we call system normality, are not readily available, which thus make the detection task deem to be $NP$-hard.

Actually, anomaly detection can be regarded as an induction problem in some sense. Assume that we have an unordered set of $n$ finite description of observable events (strings of symbols), $O_1, O_2, O_3, \cdots, O_n$. Given a new event at time $t$, $O_t$, what is the probability that it belongs to the set? A well fitting anomaly detector with good description for the known set of events is expected. The universal distribution [23] gives a criterion for goodness of fit of such description. According to our definition, the universal distribution $D_{\tilde{AD}}$ for anomaly detector $\tilde{AD}$ can be regarded as a weighted sum of all finitely describable probability measures on finite events:

$$D_{\tilde{AD}}([O_i]) = \sum_j \beta_j \prod_{i=1}^{t} p_j(O_i). \tag{1}$$

$t$ is the time step representing the number of available observation set $[O_i]$, $\beta_j$ can be taken as the weight of the $j^{th}$ probability distribution on finite observations, and its definition based on the particular detection model, for example, for an anomaly detector using string match method, $\beta_j = 1$, if ongoing events match the exact pattern $\phi$ that stored in normal pattern set $\Phi$. Suppose that $[O_i], i = 1, 2, \cdots, t$ is a set of $t$ observations generated by stochastic process $h(x)$, the probability that $D_{\tilde{AD}}([O_i])$ assigns to a new observation $O_{t+1}$ is

$$Pr(O_{t+1}) = D_{\tilde{AD}}([O_i] \bigcup O_{t+1})/D_{\tilde{AD}}([O_i]).$$

The probability assigned to $[O_i]$ by stochastic generator $h(x)$ is

$$h([O_i]) = \prod_{i=1}^{t} h(O_i). \tag{2}$$

In an effective anomaly detection model, for a suitable set of observations $[O_i]$ that used for characterizing system normality, the probability assigned by $D_{\tilde{AD}}$ in Equation (1) should be very close to those generated by hidden stochastic process $h(x)$ in Equation (2), that is, a maximal prior information an anomaly detector can posses is

the exact knowledge of $\lambda$, but in many cases the true generating process $h(\cdot)$ is not known, what we expect is that an anomaly detector based on $D(\cdot)$ performs well with small expected errors between $D(\cdot)$ and $\lambda$. For such two probability distributions on finite number of observations, a corollary derived from Hutter [11] can be given as:

**Corollary 1.** *The expected value of the sum of the squares of the differences in probabilities assigned by the stochastic generator $h(\cdot)$, and anomaly detector $D(\cdot)$ to the elements of the observation are less than a certain value, and the expected error in probability estimate might decreases rapidly with growing size of the normal data set.*

The corollary guarantees theoretically that predictions based on $D(\cdot)$ are asymptotically as good as predictions based on $\lambda$ with rapid convergence. Any *a priori* information that can be insert into $D(\cdot)$ to obtain less errors, and we believe that if all of the needed *a priori* information is put into $D(\cdot)$, then (1) is likely to be the best probability estimate possible to $h(\cdot)$, and thus anomaly detector could achieve one hundred percent accuracy. So far, neither modelling approaches, which aim to estimate $c$, $n$,$m$, nor nonmodelling approaches, which deduce and generate normal behavior rules using heuristic, clustering algorithms, data mining techniques and statistical measures, have given a thorough solution. Actually, the limited samples we can obtain, together with corresponding sampling errors, determine what we can do is just estimate and predict system normality in an approximate way.

## 2.2 Sequence-based Analysis

In many cases, the ordering property rather than the frequency property dominates the characteristic of observable subjects, the pattern of $Set(O_t, w)$ rather than the individual event $O_t$ is thus of potential interest, and the ongoing events should be considered in a consecutive manner instead of independently. Based on the assumption that current event $O_t$ is related with previous events, hidden generation process, and time instant $t$, a pair of probability distribution can be given as following:

$$Pr\{O_t|H(t)=1, O_{t-1}O_{t-2}\cdots O_1, t\}$$
$$Pr\{O_t|H(t)=0, O_{t-1}O_{t-2}\cdots O_1, t\}$$

for most problems, the ultimate goal is just to identify a short temporal pattern of anomalous events, therefore, the sequence $O_{t-1}O_{t-2}\cdots O_1$ can be replaced by $Set(O_t, w)$,

$$Pr\{O_t|H(t)=1, Set(O_t, w), t\}$$
$$Pr\{O_t|H(t)=0, Set(O_t, w), t\}.$$

Similar to the analysis for unordered event set, a posterior probability of anomaly detection based on temporal-related events can be given as:

$$\begin{aligned} & Pr\{H(t)=1|O_t, Set(O_t, w), t\} \\ = & \frac{Pr\{O_t|H(t)=1, Set(O_t, w), t\}\cdot(1-\lambda^{'})}{Pr_3\cdot(1-\lambda^{'})+Pr_4\cdot\lambda^{'}} \\ = & \frac{c\cdot(1-\lambda^{'})}{c\cdot(1-\lambda^{'})+\lambda^{'}} \\ & Pr_3 = Pr\{O_t|H(t)=1, Set(O_t, w), t\} \\ & Pr_4 = Pr\{O_t|H(t)=0, Set(O_t, w), t\}. \end{aligned}$$

Where $\lambda^{'} = Pr\{H(t)=0, Set(O_t, w), t\}$ is similar with $\lambda$, represents a *p*riori probability of the legitimate pattern which contains $w$ consecutive events that has been generated by $h(x)$, and an unknown constant

$$c = \frac{Pr\{O_t|H(t)=1, Set(O_t, w), t\}}{Pr\{O_t|H(t)=0, Set(O_t, w), t\}}$$

From the above formulation, we do not know with certainty the generation of $Set(O_t, w)$ by mixture process $h(x)$, nor do we know the distribution of $M(t)$ and $N(t)$. The ongoing event $O_t$ may depend on the current time step $t$, as well as the temporal pattern of events generated at time steps prior to $t$, which allows the possibility that $M(t)$ and $N(t)$ are non-stationary. Furthermore, instead of restricting our attention on $Set(O_t, w)$ whether and which its subsequence is generated by $M(t)$ or $N(t)$, we regard it as a whole dynamic temporal pattern, therefore, the detection problem of interest is to decide whether the appearance of ongoing event reveal the temporal pattern includes $w$ events as anomalous, rather than concern the individual $O_t$, however, we do not exclude the possibility that the sudden appearance of anomalous event uncover any previous potential anomalies at once.

Similarly, the estimation of $Pr\{O_t|H(t) = 1, Set(O_t, w), t\}$ and $Pr\{O_t|H(t) = 0, Set(O_t, w), t\}$ can also be roughly considered as a simple inductive inference problem: *Given a string $O_{<t}$ (denote $O_1, O_2, \cdots, O_{t-1}$), take a guess at its continuation $O_t$.* Specially, the generation of the event sequence $O_1, O_2, \cdots, O_{t-1}$ is governed by a hidden stochastic process $h(\cdot)$, and $\mu$ is unknown probability distribution for taking $O_t$ at particular time instant $t$ based on the available event $O_1, O_2, \cdots, O_{t-1}$, i.e. $\mu(O_t|O_{<t})$, while $\rho$ is a guess probability distribution close to $\mu$ or converges, in a sense, to $\mu$, and we expect that an anomaly detector based on $\rho$ performs well. Assume $P := \{p_1, p_2, \cdots, p_n\}$ is a countable set of candidate probability distributions on event sequences, a universal probability distribution $\pi$ hence can be defined as:

$$\pi(O_{1:t}) := \sum_{p\in P} w_p p(O_{1:t}), \sum_{p\in P} w_p = 1, w_p > 0. \quad (3)$$

As the above notations, $P$ is known and might contain the true distribution $\mu = p_i$ if $P$ is sufficiently large or with well characterization. Based on those assumptions, two corollaries therefore can be deduced from theorems of [11] as follows for modelling anomaly detection models:

**Corollary 2.** *Convergence: Assume anomaly detector observe a sequence $O_1O_2\cdots$ over a finite space $S$ drawn with probability $\mu(O_{1:n})$ for the first n events. The universal conditional probability $\pi(O_t|O_{<t})$ of the next symbols $O_t$ given $O_{<t}$ is related to the true conditional probability $\mu(O_t|O_{<t})$ in the following way:*

$$\sum_{t=1}^{n} E_{<t} \sum_{O_t} (\mu(O_t|O_{<t}) - \pi(\mu(O_t|O_{<t}))^2 \leq \ln w_\mu^{-1}$$

*where $E_{<t}[\cdot] := \sum_{x<t\in P^{t-1}} \mu(x_{<t})[\cdot]$ is the expectation and $w_\mu$ is the weight (4) of $\mu$ in $\pi$.*

which shows that the predication accuracy of anticipated anomaly detectors are asymptotically as good as predications based on the stochastic generator $h(\cdot)$ with rapid convergence. However, in practice, ongoing observation might not have exact matching pattern in $P$, i.e., $\mu \notin P$, in such case, a "nearby" distribution $\hat{\mu}$ with weight $w(\hat{\mu})$ is expected, and the distance between $\hat{\mu}$ and $\mu$ is bounded by a constant. The convergence of anomaly detectors determines the amount of training time or data required to have a stable model, and the detector converges well when most of the "anticipated" patterns appear repeatedly and are extracted well.

**Corollary 3.** *Error Bound: Assume anomaly detector observe a sequence $O_1O_2\cdots$ over a finite space $S$ drawn with probability $\mu(O_{1:n})$ at time $t$. $\Theta_\pi$ is the universal prediction scheme (used by probabilistic anomaly detector $\tilde{AD}$ to determine the deviation between normal sequence and abnormal ones) based on the universal prior $\pi$, $\Theta_\mu$ is the optimal prediction scheme based on the stochastic generator $h(\cdot)$. The total u-expected number of prediction errors $E_n^{\Theta_\pi}$ and $E_n^{\Theta_\mu}$ of $\Theta_\pi$ and $\Theta_\mu$ are bounded by:*

$$0 \leq E_n^{\Theta_\pi} - E_n^{\Theta_\mu} \leq \sqrt{2Q_nS_n} \leq 2S_n + 2\sqrt{E_n^{\Theta_\mu}S_n}$$

*where $Q_n = \sum_{t=1}^{n} E_{<t}$ is the expected number of non-optimal predictions made by $\Theta_\pi$, $S_n := \sum_{t=1}^{n} E_{<t} \sum_{O_t} (\mu(O_t|O_{<t}) - \pi(O_t|O_{<t}))^2$ is the squared Euclidian distance between $\mu$ and $\pi$.*

The corollary actually gives the upper bound of the false alert rate of an ideal sequence-based anomaly detector. We usually pay our attention to the lower bound of the false alert rate of anomaly detectors, but in fact, all the possible detection schemes also have a upper bound to some extent. Although it makes little sense on designing an anomaly detection system with near zero false alert rate, it really gives us an impression that any anomaly detection schemes based on sequence prediction would never perform too badly. And obviously, how to select a universal probability distribution $\pi$, specifically, $p_i \in P$ and $w_i$, is always the key to design an ideal sequence-based anomaly detection system.

Rather than considering the specific design of anomaly detectors, here we just attempt to show that anomaly detection problem essentially is also a prediction problem in some sense. Related proof of those two corollaries can be found in [11], which provides theoretic foundation for any anomaly detection scheme, and shows that probability distribution of the expected controllable process converge to that of the hidden stochastic process and limited by errors bound. Based on the historic data, the extent of the deviation between an expected event and ongoing event thus determines whether anomaly appears.

Generally, this section casts the anomaly detection problem in a statistical framework to describe the anticipated behavior of anomaly detectors from an overall viewpoint, which facilitate us to construct a basic modelling for the further discussion in the latter part this paper. Although the unrestricted assumption of the framework is quiet complex and general, it is nevertheless meaningful to provide an outline for our detailed analysis. As we know, many of subjects that anomaly detection scheme to examine are notoriously noisy, non-stationary, and defined on extremely large alphabets, while our framework extracts them to a comprehensible and manageable level, and based on which, we select several typical subjects that have been widely used for analysis.

# 3 Normality Characterization of Observable Subjects

Basically, two kinds of observable subjects from computer systems can be selected as the objects for monitoring and analyzing in order to capture the anomalous traces, namely, hosts in the network and the communication links among the hosts. From a high level view, several criteria to the selection of observable subjects need consideration, in order to characterize the system normality effectively:

- Availability, the basic condition, which means that the subject can be observed and captured directly or by some assist tools.
- Tangibility, which means that subjects can be recorded in a specific form, and can be recognized or dealt with in a particular way, such as user profiles or audit files.
- Operability, a subject might have a large number of attributes, but it should be possible to be managed by some techniques such as attribute projection, feature selection or value aggregation.
- Sensitivity, which means that the subject is both robust to variations in normal, and perturbed by intrusion, so that it can reflect the changing of system normality well.

## 3.1 Normality of the Observation from Hosts

A great number of variables could be employed to characterize the state of a host, such as command line strings

[18, 19], system call traces [5], resource consumption patterns [16], etc. The properties of all those variables could be encompassed into the framework that we established in the last section. However, in fact, the normal behavior of many variables does not have obvious pattern, which would be taken as "noise" of "normality". Burgess et al. [1] gave a careful analysis on the computer system normality, according to which, the system can be distinguished as three scales:

- *Microscopic,* details exact mechanisms at the level of atomic operations, such as the individual system calls and other atomic transactions in operating systems (in terms of *milliseconds*).
- *Mesoscopic,* looks at small conglomerations of microscopic processes and examines them in isolation, such as the individual process or session, or a group of processes executed by one program (in terms of *seconds*).
- *Macroscopic,* concerns the long-term average behavior of the whole system, such as the periodical activities of the users and their corresponding resources consuming patterns.

All the host subjects fall into these three categories, and can be taken as the objects for anomaly detectors, whether it aims to look for suspicious patterns or attempts to identify the values that deviate from the acceptable distribution of values. But actually, most of the available host-based anomaly detection methods take subjects at *mesoscopic* level due to its better controllable attributes to establish anomaly detection models. For instance, Forrest et al. [5, 8] ever proposed an immunological detection model by analyzing system calls sequences, which focus on the *mesoscopic* level of UNIX operating system, and some subsequent independent works [14, 15] also take system calls sequences as observable subjects. Consequently, the motivation to analyze the normality of the mesoscopic scale is obvious, that is, why system calls sequences can be selected as observation? What attributes these sequences have? Whether the regularity of such computing environment benefits the anomaly detection? Actually, Forrest et al. [5] has given an satisfied answer for the first question, but for the last two questions, there are still some problems need further exploration.

Most the work took the name of the system calls as the observable (other parameters passed to the system calls are ignored), after sequence is established, namely, $(s_1, s_2, \cdots, s_l)$, detection methods such as Enumerating Sequences, Frequency-based methods, Data mining techniques, HMM (Hidden Markov Model), or some text categorization methods were applied to identify anomalies. The work of Lee et al. [14] showed that additional information to the sequence elements would improve detection performance without considering the trade-off between detection accuracy and computational cost. For instance, sequence can be established as $(s_1\_o_1, s_2\_o_2, \cdots, s_l\_o_l)$ or $(s_1, o_1, s_2, o_2, \cdots, s_l, o_l)$, where $o_i$ represent the *obname* of system call $i$. Additionally, Lee et al. gave an analysis for the regularity of these objects using information-theoretic measures, such as entropy, conditional entropy, relative conditional entropy, information gain and information cost, which gives us a good clue for the characterization of the system normality. Specifically, for an audit data set $X$ where each data item belongs to a class $x \in C_x, y \in C_y$, several information theoretic measures can be used to describe its characteristics, in order to built an appropriate anomaly detection model:

- *Entropy:*

$$H(X) = \sum_{x \in C_X} P(x) log \frac{1}{P(x)},$$

where $H(X)$ is the entropy of $X$ relative to $C_X$, and $P(x)$ is the probability of $x$ in $X$. As we know, the amount of variability is most easily characterized by the entropy of the signal, if the variations in data are equally distributed about some preferred value, the distribution over a sufficient number of instances would be normal. $H(X)$ thus can be used to measure the regularity of the record in audit data, and the data set with smaller entropy would improve the detection performance due to its purer nature and simpler structure.

- *Conditional Entropy:*

$$H(X|Y) = \sum_{x,y \in C_X, C_Y} P(x,y) log \frac{1}{P(x|y)},$$

As we explained in the last section about sequence-based anomaly detection models, for two sequence sets,
$X = (x_1, x_2, \cdots, x_m)$, $x_i = (e_i^1, e_i^2, \cdots, e_i^{n-1}, e_i^n)$,
$Y = (y_1, y_2, \cdots, y_m)$, $y_i = (e_i^1, e_i^2, \cdots, e_i^{k-1}, e_i^k)$,
where $e_i^j$ represent the event and $k < n$, $H(X|Y)$ thus can be used to measure the regularity of sequential dependencies, that is, how much uncertainty remains for $e_i^{k+1} \cdots e_i^n$ of $xi$ with knowledge of $y_i$. Obviously, the smaller the values is, the more deterministic of the sequence $x$ after $y$ is obtained, which therefore benefits the build of anomaly detection models.

- *Relative Conditional Entropy:*

$$E(p|q) = \sum_{x \in C_X} p(x) log \frac{p(x)}{q(x)},$$

where $p(x)$ and $q(x)$ are two probability distributions over the same $x \in C_x$, and $E(p|q)$ can be applied to measure the similarity of two datasets (e.g. training data and test data). The distance (similarity) between two audit datasets could provide us *a prior* knowledge to build and evaluate anomaly detection models.

- *Information Gain:*

$$Gain(X, A) = H(X) - \sum_{v \in Values(A)} \frac{|X_v|}{|X|} H(X_v),$$

where $Values(A)$ is the set of possible values of $A$ and $X_v$ is the subset of $X$ where $A$ has value $v$. $Gain(X, A)$ can be used as a criteria to select important attributes for achieving better classification, and thus prediction performance, essentially, it has the similar contribution as conditional entropy to measure regularity of sequential dependencies.

Although there are still some details about the data normality worth consideration, the proposed information-theoretic measures give us some fundamental understanding about the regularity of computing environment that the anomaly detectors work. Lee et al. [14] applied conditional entropy to determine the appropriate length used for sequencing the system calls to construct an anomaly detection model with the conclusion that there is a relationship between the fall of in entropy and the appropriate window size for probabilistically-based classifiers. But interestingly, Tan et al. [24] suggested that conditional entropy is not a universal sequence-length selection metric, and it almost has the same appearance in a general manner, independent of the particular datasets, which undermines its effectiveness. However, we still believe that those information theories can contribute to the characterization of the environment normality, and thus improve the performance of anomaly detectors to some extent. Moreover, we have already found out the intersection between those information-theoretic measures and the stochastic framework we have discussed in the last section, especially for those sequence-based anomaly detection models.

To measure the computer system normality from a macroscopic level, Burgess et al. [1] applied a scaling transformation to the measured data, and the distribution of fluctuations about the mean was approximated by a steady-state, maximum-entropy distribution with modulation by a periodic variation. The idea can be brief described as:

*Motivation for Transformation:* the entropy of the collected data are computed to gauge the variability of the signal, which indicates that signal is maximally; average and standard deviations are computed in terms of periodicity, and the periodogram standard deviation is itself a pseudoperiodic functions of time, which shows that the system acts as a scale of activity that varies in time; each time is rescaled by its local standard deviation, and the scaled distribution of measurements at a given periodic time is closely resembles a Planck distribution.

*Transformation:* As the entropy to be high, processes which have "fluctuation structure" can be written in exponential form $exp(-\beta E_i)$ as a Boltzmann distribution with some arbitrary set of parameters $E_i$, which satisfies the maximum entropy condition for fitting the data; The probability distribution is approximately written as

$$p[q] = exp(-\beta E[q]) / \int dq exp(-\beta E[q]).$$

To determine parameters $E[q]$, a stochastic model is used:

$$E[q] = \int dt[(\frac{dq}{dt}^2) + V(q)],$$

As the system is moderately loaded, two simple assumptions are based on: (a) maximal entropy of data and (b) fluctuations at no cost, therefore, $V(q) = 0$. Finally, Planck distribution, which is the form of the equivalent, transformed steady-state system is yielded through computing the fluctuation spectrum for the model on a periodogram.

Burgess et al. gave a method to characterize system normality from the point of view of macroscopic scale, which inspire us to detect host-based system anomalies from a macro perspective, however, due to its approximate nature, any attacks with normal pattern appearance are difficult to be identified based on such model, in addition, what information are required and effective for detecting anomalies need further exploration, and it heavily depends on what will we do once anomalies have been discovered. Intuitively, the normality of those observable subjects from mesoscopic and macroscopic scales could be combined to achieve better performance, macroscopic normality is used to monitoring the variant of system coarsely, while mesoscopic give doubtful activities further analysis and fine-grain characterization.

## 3.2 Normality of the Network Observation

Due to the diverse nature of the computer network, it is almost impossible to establish an ideal mathematical model with perfect characterization of the normality of observable subjects, i.e. network packets, nor it is easy to design efficient intrusion detection techniques for networking. However, this does not only for intrusion detection, but also more or less for other fields, such as traffic modelling and analysis. In this sense, the fundamental understanding of basic protocol behavior is a possible way to go. In addition, due to the inherent limits of the available IDSs and the increasing application of encryption in communication, such as IPSec, SSL, intrusion detection and prevention have once again moved back to the host systems. Here, we only propose some preliminary ideas to measure network normality, while further experimental analysis and verification are left to our later work.

So far, *tcpdump data* has been widely applied to detect attacks from the protocol scale (connection behavior). Generally, each record describes a connection using several features: timestamp, duration, source port, source host, source bytes (outbound bytes from source to destination), destination port, protocol type(TCP, UDP,

ICMP or others), destination host, destination bytes (inbound bytes from source to destination), and flag. Due to the huge data amount generation everyday and the transient nature, it is really difficult to describe the system normality in details, and therefore simplification and preprocess is needed. Taking those features as various attributes, Lee at al. [14] used information gain as guiding principle to partition *tcpdump* data based on the assumption that the smaller the entropy is, the more regularity the dataset, and therefore benefit for modelling and characterizing anomaly detectors, and conditional entropy was applied to compute temporal and statistical features. Although it is true that such pre-analysis could facilitate anomaly detection modelling, huge amount of data and transient nature make it is time-consuming to determine the proper granularity of the subjects. Some techniques for online analysis of continuous stream give us some clues to capture the transient nature of network subjects [2, 6]. Additionally, some network traffic modelling methods also give us some inspiration to monitor and obtain the necessary information for measuring network normality at a macroscopic level [17].

In order to develop a traffic model which can accurately characterize the diverse statistical properties with complex temporal correlation and non-Gaussian distributions of heterogeneous network, Ma et al. [17] proposed a wavelet domain-based models. In these models, correlation structures of wavelet coefficients for long/short-rang dependence processes are reduced to only a few key elements. For Gaussian traffic, Markov models can be implemented through a linear model on wavelet coefficients to capture the short-range dependence among wavelet coefficient, i.e.

$$d_s = \sum_{l=1}^{s-1} a_s(l) d_l + b_s w_s, 1 \le l \le N$$

where $a_s(l)$ and $b_s$ are weighting factors depending on the one-dimensional index $s$, and $w_s$ is i.i.d Gaussian noise with zero mean and a unit variance. The value of $s$ and $a_s(l) = 0$ determines the model and the relations between wavelet coefficients, for example, when $s = 1$, and $a_s(l) = 0$ for all $l$, the model is the simplest one, i.e., an independent wavelet model.

For non-Gaussian distribution traffic, a shaping algorithm was derived using the relationships among wavelet coefficients, scale coefficients, and the cumulative process. Specifically, it includes two stages:

- *Traffic Modelling:* wavelet transform on a training sequence $\hat{x}$ to obtain wavelet coefficients and scaled coefficients, and then estimate the variance of wavelet coefficients and the cumulative probability function of scale coefficients at each time scale.

- *Synthetic Traffic Generation:* generating the background wavelet coefficients by Gaussian wavelet model and compute the shaped wavelet coefficients and scale coefficients recursively for all time scales, after wavelet inverse transformation, synthetic sequence $\tilde{x}$ is obtained.

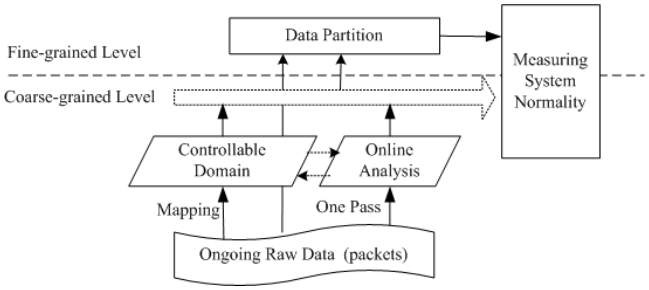


Figure 1: A simple framework for measuring network normality

Therefore, after wavelet transformation, whatever short- and long-range temporal dependence traffic are all "short-range" dependent on the wavelet-domain, which facilitates significantly the characterization of network normality and our analysis of anomalies at a macro level.

The countermeasure to deal with the transient nature of network observable subjects is online analysis, that is, process the data in a single pass, or a small number of passes. For instance, under some definition of "similarity", similar items can be clustered in the same partition, while different items are in different partitions. Based on the existing *facility location algorithm*, Guha et al. [6] modified it to produce exactly $k$ clusters for solving k-Median problem in one pass, their experiment on KDD-CUP 99 intrusion detection data showed that raw tcpdump could be clustered into five clusters with 34 continuous attributes. In addition, Cormode et al. [2] ever proposed a novel algorithm for calculating a small summary for any data stream, i.e. $l_o$ *sketch*, and employed *Hamming norm* to estimate the similarity of streams online, which also give us a rapid and ease method to analyze network regularity.

Based on the available techniques we have analyzed, a framework for measuring network normality can be concluded through a top-down procedure as follows (its skeleton is shown in Figure 1):

1) *Coarse-grained Level:*

   - Mapping network traffic into wavelet domain to discover the periodicity of the specific network activities, which can disclose the sudden system collapse and unrhythmic activities;
   - Sketch-based techniques and clustering methods are applied to a certain doubtful time-scale (or a periodicity) to have further insightful investigation.

2) *Fine-grained Level:*

   - Information-theoretic measures are used to divide the processed network data from coarse-grained level into more "pure" data sets with higher regularity;

- Building anomaly detection models based on the characterization of system normality.

Actually, collection and monitor of network observable subjects in a discrete way rather than a continuous way may not deteriorate the performance [1]. From the point of view of the observable subjects, we envision a framework in which several levels of data analysis are used as the basis to be combined to yield a single but effective system normality characterization. We envision further an approach in which anomaly detection models are built on the fundamental understanding of their operating environments, and have the adaptability in response to changing situation. The hope is that a collection of simple, elaborate surrogates based on specific observable subjects can evolve into generic models without performance deterioration. From the similar motivation, a host-based autonomic detection coordinator have been developed in [31].

# 4 Case Studies

Generally, operating environment means the working situation constructed by the observable subjects that anomaly detectors working with, and most of them can be cast in the framework we proposed in Section 2. In the last section, we gave a general discussion of normality characterization for observable subjects from hosts and network. After a broad survey of the existing literature on anomaly detectors, we found that most work pay more attention to the design of the anomaly detection models themselves, rather than the operating environment. Here, we take two kinds of anomaly detectors (frequency-based and sequence-based ) as instances to insight their operational mechanisms from the perspective of operating environment.

## 4.1 STIDE Detector

The stide algorithm can be described as follows [5]:
*Predefinition:* for two sequence $X$ and $Y$,
$X = (x_1, x_2, \cdots, x_N)$, $Y = (y_1, y_2, \cdots, y_N)$,
the similarity between them is defined as:

$$Sim(X,Y) = \begin{cases} 0 & \text{if } x_i = y_i, \text{ for all } i,\ 0 \le i \le (N-1) \\ 1 & \text{otherwise} \end{cases}$$

*Given* a set of sequences in the normal database, $\{Y_1, Y_2, Y_3, \cdots, Y_M\}$, $|Y_i| = N, 1 \le i \le M$, and a ordered set of sequences in test data, $\{X_1, X_2, X_3, \cdots, X_{Z-(N-1)}\}$,
where $X_s = (x_s, x_{s+1}, \cdots, x_{s+(N-1)})$ for $1 \le s \le (Z-(N-1))$, and the size of test data is $Z$, the similarity measure assigned the sequence $X_S$ is:

$$\hat{Sim}(X_s) = \begin{cases} 1 & \text{if } Sim(X_s, Y_j) = 1, \text{ for all } j,\ 1 \le j \le M \\ 0 & \text{otherwise.} \end{cases}$$

Finally, locality frame count (LFC) with size $L$ for each size $N$ sequence in the test data is defined as:

$$LFC(X_s) = \begin{cases} \sum_{l=((s-L)+1)}^{s} \hat{Sim}(X_l) & \text{for } s \ge L \\ \sum_{l=1}^{s} \hat{Sim}(X_l) & \text{for } s < L \end{cases}$$

Based on this algorithm, a concise database containing normal sequences with length $N$ can be generated for detecting anomalies. The algorithm is easy and effective, some more sophisticated models do not have significant performance improvement over the original model [26]. In the original work, the sliding window of the STIDE detector was set 6, Lee at al. [14] gave an analysis using *conditional entropy* to explain the selection of the "magic number", but Tan et al. [24] undermined the entropy-based analysis using a random data set. Furthermore, they gave a thorough analysis on the selection of detector window using a synthetic data set [24, 25]. Actually, this phenomena depends heavily on the STIDE's operating environment, and the detector essentially works in an exhaustive way, its performance therefore is effected by the normal data set, any foreign elements or sequences that unincorporated in the normal data set would be detected easily. As Maxion et al. [18] analyzed, STIDE has a blind region under $x = y$ in coordinate, where x-axle represents "size of foreign-sequence anomaly" and y-axle denotes "size of detector window". The existence of blind region cause the detector to suffer from simple exploits by a sophisticated attacker who have fundamental understanding with its operational limits. Therefore, the analysis and construction of normal sequence data set is essential to improve the performance of STIDE. The tradoff between the cost and accuracy is the variant detector window above six.

## 4.2 Minimum Cross Entropy-based Anomaly Detector (MCE)

Based on the assumption that the occurrence frequencies of different observable subjects can be measured during a certain time scale, a probability distribution can be used to represent the occurrence pattern during this period. In this model, the sequential property is out of consideration, which essentially is a kind of static method [4]. The method has not been widely used because of its unsatisfactorily performance in some situation. Its basic idea can be described as follows:

Assume $P(M)$ denotes the probability distribution characterizing the behavior of a normal model $M$ and $P_i(M)$, $i = 1, 2, \cdots, N$ denote the occurrence probability of event $i$ among a set of $N$ events, the similarity of two distributions $P$ and $Q$ can be measured using *cross entropy*:

$$\begin{cases} C(P,Q) = \sum_{i=1}^{N} (Q_i - P_i) log \frac{Q_i}{P_i}. \\ C(P,Q) \ge 0, \\ C(P,Q) = 0 \Leftrightarrow P = Q. \end{cases}$$

After determining a threshold for the similarity between $P$ and $Q$ using training data and validation data

set, we can decide whether ongoing events set should be considered as intrusive with respective to the normal model. Actually, the performance of this method might be improved significant with the preprocess of data using *information-theoretic* measures that we discussed in last section.

Here, we do not intend to undermine the contribution of the work [4], and we only want to point out that a careful analysis of the operating environments that anomaly detectors work could also obtain the same conclusion as that from expensive *trial-and-prone-to-error* experiments. In their work, the anomaly detector operated with two kinds of observable subjects, one is program profiles based on Unix system calls, another is user profiles based on Unix shell commands. As we know, system calls executed by the same process have certain temporal pattern, namely, system calls from a specific process have the sequential correlation, at least the order between several system calls always keep unchanging. While for the shell command data, although individual user has particular pattern during his/her login session, that is, the token was recorded almost always keep the same entropy, the frequency of tokens rather than the sequential relations have more contribution to the characterization of user behavior. Under such cases, anomaly detectors which can capture temporal characteristics, such as HMM-based anomaly detector, obviously have better performance in the system calls data set than that of in the shell command data set. On the contrary, frequency distributions-based anomaly detector have the inverse performance due to the properties of operating environment. Therefore, after simply but effective analysis of the operating environments, we can get the same conclusion that [4] ever got easily.

## 4.3 Probabilistic Anomaly Detectors

Ye et al. [27] gave a nearly thorough analysis on the probabilistic techniques-based anomaly detectors with computer audit data, including decision tree, Hotelling's $T^2$ test, chi-square multivariate test and Markov chain. Part of conclusion they obtained was "*...unless the scalability problem of complex data models taking into account the ordering property of activity data is solved, intrusion detection techniques based on the frequency property provide a viable solution that produces good intrusion detection performance with low computational overhead.*"

Among the various probabilistic techniques-based intrusion detectors, expect Markov chain, all the others can be regarded as static intrusion detectors due to their statistical nature (although some ordering property of the observable subjects were also considered). Our analysis on their operating environment is motivated by following questions:

- *Whether* the property of the selected observable subjects have been explored thoroughly?
- *If not*, whether complex models could discover more? *otherwise*, whether frequency property is enough for their operational performance?
- Can we get a conclusion that some information will be lost when only event type of computer audit data are used to characterize system normality?

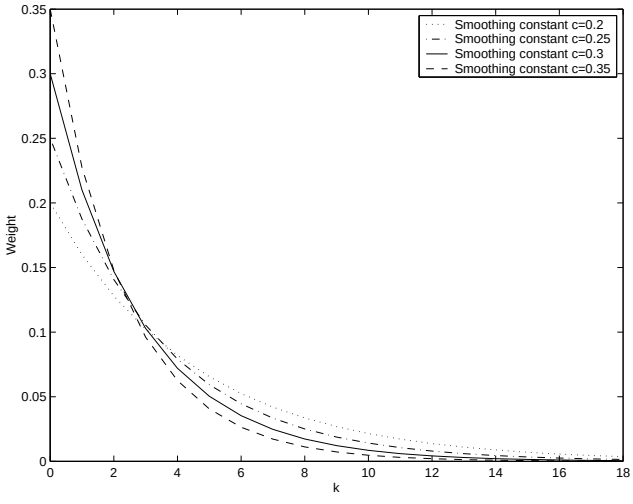


Figure 2: Decay effect with different smoothing constants

Here, we only consider the basic data model that all the probabilistic anomaly detectors applied. In the model, the observable subjects, namely, *audit data* are represented as frequency distribution $(X_1, X_2, X_3, \cdots, X_N)$, where $N$ denotes the number of different event in the audit set, and the exponentially weighted moving average method (EWMA) was applied to compute the value of $X_i$, specifically, if the current event $t$ belongs to the $i$th event type,

$$X_i(t) = c * 1 + (1 - c) * X_i(t - 1),$$

if the current event $t$ different from the $i$th event type,

$$X_i(t) = c * 0 + (1 - c) * X_i(t - 1),$$

where $X_i(t)$ is the observed value of the $i$th variable in the vector of an observation $(X_1, X_2, X_3, \cdots, X_N)$ for the current event $t$, thus a $M \times N$ vector with $M$ target values is constructed if the observation set has $M$ data points; $c$ is the smoothing constant that determines the decay rate; and $1 \leq i \leq N$. This model can convey not only the relative frequency distribution of $N$ in a sequential events during a certain time scale, but also reflect the intensity of activities. However, from the point of view of the observable subjects, two aspects of the data modelling worth insightful consideration, i.e., the selection of parameter $\lambda$, and the correlation among data points. A figure below shows the decay effect of different smoothing constants.

We can see from Figure 2 that after a certain period, the weights drops close to zero, but the speed is different due to the various value of $c$. For example, when $c = 0.3$, the frequency value of $X_i(t)$ at the current event

considers about the past 15 audit events ($k = 0, \cdots, 14$), while past 22 events ($k = 0, \cdots, 21$) are taken into account when $c = 0.2$. In work [28], $c$ was set to 0.3−a commonly used value for the smoothing constant, other values were not tried and compared. Although we do not expect that some unknown $c$ could improve the modelling performance dramatically, a comparative study should be carried out to insight the impacts of different values, and thus select one for better modelling. Furthermore, $c$ might vary in different situation, due to the drifting of system normality, a constant value thus can hardly characterize all the normal activities well.

Both the normal and intrusive training data can be represented using the frequency distribution representation, and thus probabilistic techniques such as $Hltelling's\ T^2\ Test$, $Chi - Square\ Multivariate\ Test$ can be used to calculate the distance between testing data and training data. An assumption to support this model is that testing data are taken as a whole collection of audit events. Although some ordering property is carried in the model, the knowledge of the unobservable process distribution is ignored. As we know, each process might generate a group of audit events, and there might exist some intervals between those groups, an underlying continuous measurement therefore should be considered in the data model, in order to capture the process shift. Based on this fact, a grouped data EWMA model [9], rather than variables based EWMA, might have more contribution to the characterization of computer audit events.

Additionally, in the original data model, only the audit event type was considered, while other attributes, such as user ID, process ID, session ID, the system object accessed, were omitted. To incorporate those necessary additional information, a multivariate EWMA can be used as follows:

$$X_i(t) = C * O_i(t) + (1 - C) * X_i(t - 1),$$

where $X_i(t)$ is the $i$th EWMA vector, $O_i(t)$ is the $i$th observation vector at time $t$, $i = 1, 2, 3, \cdots, n$, $C$ is the diag $(c_1, c_2, \cdots, c_p)$ which is a diagonal matrix with $c_1, c_2, \cdots, c_p$ on the main diagonal, and $p$ is the number of variables, i.e., the number of attributes that we are considering. The MEWMA model takes into account all the necessary variables of audit events, and thus can be used to capture the process shift in multi-scales. Although it is much more complex than the univariate EWMA, a better performance is expected to be achieved if some scalability problems are solved well.

Preliminary analysis shows that the characterization of the operational situation has great effect on the anomaly detector's performance. Tracing back to the problems posed in the beginning of this subsection, we infer that more accurate/complex data models might benefit the improvement of anomaly detector's detection performance. However, scalability problem is another obstacle, which was claimed in [29]. The work also proved that the performance of first-order Markov chain is better than that of high-order stochastic models, although the latter one has more complex model (means more expensive computational cost) than the former one.

## 4.4 Comparative Analysis

As former analysis, all the anomaly detectors are specialized by their different detection coverage or blind spot, part of which attribute to operating environment. We hope that a thorough comparison analysis could provide us an approach to combine anomaly detectors together to achieve a broader detection coverage. In fact, the statistical modelling in Section 2 facilitates the comparison between those anomaly detectors, in terms of detection capability and operational limits.

A brief compared results is shown in Table 1, where $N$ is the size of normal data set that has been constructed in a particular form, while $L$ is the size of ongoing trace being detected. For STIDE, $w$ is a predefined window size. What we compare here is only the detection cost, while the cost of models' construction are not considered. Note that the detection cost of STIDE can be reduced to $L * logN$, if normal data are stored in an effect form, i.e., forest of trees. The detection cost of probabilistic detectors are differ in specific techniques, for instance, $Hotelling's\ T^2$ requires a large memory to store the variance-covariance matrix and much time to compute the matrix multiplication and inverse, its time complexity for detection nearly $O(N^2)$ ($L << N$), while $Markov\ chains$ or $chi\text{-}squre$ multivariate test need less computational overhead, i.e., $O(N)$ or so.

Although the original detection models have their own operating environments. Careful analysis allow them to be extended to a broader application field. For example, STIDE was originally developed with system calls of privileged programs, but it can also be applied to audit events provided the scope of activities is not so wide, based on the similar properties of those two observations. Similarly, the probabilistic anomaly detectors that were originally operated with audit events and shell command lines can also be extended to system calls, if enough ordering property are included during the data modelling.

Among those detectors, STIDE has the highest detection capability in general case, because it stores all the unique system calls sequences in the normal profile. Any ongoing traces with system call sequences that never appeared in normal profile will be detected as anomalies (determined by LFC). According to Corollary 2, STIDE has a good convergence due to the high average value of $w_\mu$. Generally, two elements contribute to the higher detection capability of STIDE:

- Observable subjects, i.e., system calls. As we know, system calls of privilege processes is a good level to reflect the user behaviors due to its limited range of actions, sensitivity to changes, and stability over time. While shell user command lines and audit events have less characteristics compared with system calls.

Table 1: A comparison between three typical anomaly detectors

| Anomaly Detector | | Observation | Main Property | | Detection Cost |
|---|---|---|---|---|---|
| | | | Frequency | Ordering | |
| STIDE | | System calls | | √ | $N*(L-w+1)$ |
| MCE | | System calls/ User commands | √ | | $N*L$ |
| Probabilistic Detectors | Markov Chain | Audit Events | √ | √ | $N*L$ |
| | Hotelling's $T^2$ | | | | $N^2*L$ |
| | Chi-square | | | | $N*L$ |

- Nearly exhaustive searching mode. All the available unique system calls sequences are used to characterize system normality, which constructs a broad boundary to encompass normal behavior.

For frequency-based anomaly detectors, less characteristics of their operating environments, e.g., unpredictable range of activities, instabilities over time, cause them to suffer from low detection capability. According to the Corollary 1, only the huge size of normal data set provides them an opportunity to decrease expected error between probability estimation and stochastic generator to a low level.

# 5 Evaluation of the Anomaly Detectors

Another hard stone in the anomaly detection research community is the anomaly detectors' evaluation. Most of existing IDSs take 1998 and 1999 DARPA Intrusion Detection System Evaluations Data Set [21] as benchmark for evaluating their performance, and most researches focus on tallying with detection accuracy and false positive rate of detection methods, rather than the fundamental understanding of evaluation environment. Therefore, the specific design of anomaly detectors based on particular situation, together with some strong assumptions limit their application to a broader application scope.

Mchugh [20] gave a thorough analysis of so-called benchmark data set, and proposed the essential conditions that ideal measurements should have. Briefly, it includes:

- The primary method, i.e ROC (Receiver Operating Curve), to present the results of the evaluation provides no insights into the root-causes for IDS performance, and the more helpful metrics should be developed.

- The curse of the false alarms generation has not been explained clearly, therefore, the useful description of the difference between activities that are identified correctly as an attack and those that provoke a false alarm needs more insightful investigation.

- To make sure that the false alarm rate for synthetic data has an obvious relationship to that of real data, background traffic data characterization is needed for calibrated artificial test data sets.

Up to now, we have not found such work that meet above requirements completely. With the problem that whether the environment regularity has effect on the probabilistic algorithms-based anomaly detectors, Maxion and Tan [19] provided an idea for successful data synthesis, and the result verified their hypothesis. But their model is too simple to interpret more complex anomaly detection models, and some additional observational work from real data is needed. In addition, only juxtapositional anomalies was considered in that model, while temporal anomaly detection was left.

Inspired by those former works, we have a primary idea to generate synthetic data for the general evaluation of anomaly detectors. Although it is still during the process of implementation and verification, we believe that it will contribute to the development of anomaly detection evaluation to some extent.

*Firstly*, collect pure real normal data source from a real environment, and mapping those collected data into controllable domain (for example, mapping network packets into wavelet domain and approximate host audit data as the Planck distribution respectively).

*Secondly*, apply some candidate anomaly detectors to the controllable data set, and analyze the data that ever provoked false alarms. This step should be done recursively to prune the data as pure normal data without confused false alarms.

*Thirdly*, in order to ensure the regularity of processed data, information-theoretic measures could be used to divide the data as smaller but purer ones.

*Finally*, artificial anomalies (such as foreign symbols or sequences, and rare sequences) are incorporated into the data. One way to make it more effective is to add predefined anomalies one by one, until to a determined amount.

# 6 Concluding Remarks

This work aims to explore following questions and provide some potential solutions:

- The operational limits of some anomaly detectors are due to themselves or the particular operational environments they run.
- Whether a better characterization of system normality can improve the performance of anomaly detectors (sometimes obviously, sometimes may not).
- How to select proper anomaly detectors for a specific situation when we take into count the trade-off between performance and cost.
- It is usually hard to find a general way to evaluate existing anomaly detector's performance (including those state-of-the-art ones) in terms of admitted criteria (hits, misses, and false alerts). ROC is generally regarded as a typical but superficial analysis tool.

Those questions have been analyzed and discussed in a general way based on the available achievements, although there are still some problems worth further consideration, and some proposed ideas remains verification and implementation, we believe that future work along this way could contribute additional insight for the research and application of anomaly detectors. Someone may argue that our work are obvious and straightforward, we believe that it is important to develop a framework for the anomaly detection field, including characterization, identification and evaluation of their operating environment in order to guarantee their formal and rapid development, and it seems more important than just pruning detector itself regardless of its insightful understanding and broader application. Obviously, our future work includes the implementation of our proposed ideas, and the further analysis for the operating environment of several anomaly detectors from the view of observable subjects.

# References


[1] M. Burgess, H. Haugerud, and S. Straumsnes, "Measuring system normality," *ACM Transactions on Computer Systems*, vol. 20, no. 2, pp. 125-160, May 2002.

[2] G. Cormode, M. Datar, P. Lndyk, and S. Muthukrishnan, "Comparing data streams using hamming norms (How to zero in)," *IEEE Transaction on Knowledge and Data Engineering*, vol. 15, no. 3, pp. 529-540, May/June 2003.

[3] V. N. P. Dao and V. R. Vemuri, "A performance comparison of different back propagation neural networks methods in computer network intrusion detection," *Differential Equations and Dynamical Systems*, vol. 10, no. 1&2, pp. 201-21, Jan.&Apr., 2002.

[4] Y. D. Yan and Y. Ding, "Host-based intrusion detection using dynamic and static behavioral models,' *Pttern Recognition*, vol. 36, pp. 229-243, 2003.

[5] S. Forrest, S. A. Hofmeyr, and T. A. Longstaff "A sense of self for UNIX processes," in *proceedings of 1996 IEEE Symposium on Security and Privacy*, Los Alamitos, CA: IEEE Computer Society Press, pp. 120-128, 1996.

[6] S. Guha, A. M. Meyerson, N. Mishra, R. Motwani, and L. O'Callaghan, "Clustering data streams: Theory and practice," *IEEE Transaction on Knowledge and Data Engineering*, vol. 15, no. 3, pp. 515-528, May/June 2003.

[7] P. Helman and G. Liepins, "Statistical foundataions of audit trail analysis for the detection of computer misuse," *IEEE Transaction on Software Engineering*, vol. 19, no. 9, Sep. 1993.

[8] S. A. Hofmeyr, S. Forrest, A. Somayaji, "Intrusion detection using sequences of system calls," *Journal of Computer Security*, vol. 6, no. 3, pp. 151–180, 1998.

[9] S. H. Steiner, "Grouped data exponentially weighted moving average control charts," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 47, no. 2, pp. 203-216, 1998.

[10] W. Hu, Y. Liao, and V. R. Vemuri, "Robust support vetor machines for anomaly detection in computer security," *The 2003 International Conference on Machine Learning and Applications (ICMLA'03)*, Los Angeles, California, pp. 168-174, June 2003.

[11] M. Hutter, "Optimality of universal Bayesian sequence prediction for general loss and alphabet," *Journal of Machine Learning Research*, vol. 4, pp. 971-1000, 2003.

[12] T. Lane, and C. E. Brodley, "An empirical study of two approaches to sequence learning for anomaly detection," *Machine Learning,* vol. 51, pp. 73-107, 2003.

[13] W. Lee, and S. J. Stolfo, "Data mining approaches for intrusion detection," in *Proceedings of the 7th USENIX Security Symposium*, 1998.

[14] W. Lee and D. Xiang, "Information-theoretic measures for anomaly detection," in *IEEE Symposium on Security and Privacy*, IEEE Computer Society Press, pp. 130-143, Los Alamitos, Oakland, California, 14-16 May, 2001.

[15] Y. Liao and V. R. Vemuri, "Use of K-Nearest neighbor classifier for intrusion detection," *Computers & Security*, vol. 21, no. 5, pp. 439-448, Oct. 2002.

[16] T. F. Lunt, "IDES: An ingelligent system for detecting intruders," in *Proceedings of the Symposium: Computer Security, Threat and Countermeasures*, Rome, Italy, pp. 30-45, 1990.

[17] S. Ma, and C. Ji, "Modeling heterogeneous network traffic in wavelet domain," *IEEE/ACM Transactions On Networking*, vol. 9, no. 5, pp. 634-649, Oct. 2001.

[18] R. A. Maxion, and K. M. C. Tan, "Anomaly detection in embedded systems," *IEEE Transaction on Computers*, vol. 51, no. 2, Feb. 2002.

[19] R. A. Maxion, and K. M.C. Tan, "Benchmarking anomaly-based detection systems," in *Proceedings of International Conference on Dependable Systems and Networks (DSN2000)*, pp. 623-630, 2000.

[20] J. Mchugh, "Testing intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln

laboratory," *ACM Transactions on Information and System Security*, vol. 3, no. 4, pp. 262-294, Nov. 2000.
[21] MIT Lincoln Laboratory, http://www.ll.mit.edu/IST/ideval/data /data_index.html.
[22] B. V. Nguyen, *An Application of Support Vector Machines to Anomaly Detection*, Research in Computer Science – Support Vector Machine, Course Report, Ohio University, Fall 2002.
[23] R. J. Solomonoff, *Three Kinds of Probabilistic Induction: Universal Distributions and Convergence Theorems*, http://world.std.com/ rjs/pubs.html, June 2003.
[24] K. M. C. Tan and R. A. Maxion, ""Why 6" defining the operational limites of stide, an anomaly-based intrusion detector," in *Proceedings of the 2002 IEEE Symposium on Security and Privacy (S&P'02)*, pp. 188-201, 2002.
[25] K. M. C. Tan, K. S. Killourhy, and R. A. Maxion, "Undermining an anomaly-based intrusion detection system using common exploits," *RAID 2002*, LNCS 2516, pp. 54-73, Springer-Verlag, 2002.
[26] C. Warrender, S. Forrest, and B. Pearlumtter, "Detecting intrusions using system calls: Alternative data models," *1999 IEEE Symposium on Security and Privacy*, pp. 133-145, 1999.
[27] N. Ye, X. Li, Q. Chen, S. M. Emran, and M. Xu, "Probabilistic techniques for intrusion detection based on computer audit data," *IEEE Transaction on Systems, Man, and Cybernetics-Part A:Systems and Humans*, vol. 31, no. 4, July 2001.
[28] N. Ye, S. M. Emran, Q. Chen, and S. Vilber, "Multivariate statistical analysis of audit trails for host-based intrusion detection," *IEEE Transaction on Computers*, vol. 51, no. 7, pp. 810-820, July 2002.
[29] N. Ye, T. Ehiabor and Y. Zhang, "First-order versus high-order stochastic models for computer intrusion detection," *Quqlity and Reliability Engineering Internation*, vol. 18, pp. 243-250, 2002.
[30] Z. Zhang and H. Shen, "Application of online-training SVMs for real-time intrusiondetection with different considerations", *Computer Communications*, Elsevier Science, vol. 28, no. 12, pp. 1428-1442, July 2005.
[31] Z. Zhang and H. Shen, "Constructing multi-layer boundary to defend against intrusive anomalies: An autonomic detection coordinator," in *Proceedings of the International Conference on Dependable Systems and Networks(DSN2005)*, Yokohama, Japan, pp. 118-127, June 28-July 1, 2005.

**Zonghua Zhang** received the BSc in infromation science and MSc in computer science in 2000 and 2003 respectively, both from Xidian University, Xi'an, China. He is currently a PhD candidate at school of information science, Japan Advanced Institute of Science and Technology (JAIST). His research interests include network security, information assurance/security, and intrusion detection/prevention.

**Hong Shen** received his B.Eng. degree from Beijing University of Science and Technology, M.Eng. degree from University of Science and Technology of China, Ph.Lic. and Ph.D. degrees from Abo Akademi University, Finland, all in Computer Science. He is currently a professor in Japan Advanced Institute of Science and Technology. Prior to join JAIST, he was Professor of Computer Science at Griffith University, Australia. His main research interests lie in parallel and distributed computing, algorithms, high performance networks, data mining and multimedia systems. He has published over 200 papers, with more than 80 papers in international journals including a variety of IEEE and ACM transacations. He has served on editorial boards of 7 international journals, and chaired several international conferences. Professor Shen is a recipient of 1991 National Education Commission Science and Technology Progress Award and 1992 Sinica Academia Natural Sciences Award.

**Yingpeng Sang** received the BS degree in 2001 and the ME degree in 2004, both from the Department of Computer and Communication Engineering, Southwest Jiaotong University, Chengdu, China. He is currently a second-year PhD candidate in the Department of Information Science, Japan Advanced Institute of Science and Technology. His research focuses on network security, intrusion detection, and privacy preserving computation.