POSTER: Revisiting Anomaly Detection System Design Philosophy

Ayesha Binte Ashfaq[1], Muhammad Qasim Ali[2], Ehab Al-Shaer[2] and Syed Ali Khayam[3]
[1] School of Electrical Engineering and Computer Science
National University of Sciences and Technology, Islamabad, Pakistan
[2] Department of Software and Information Systems
University of North Carolina Charlotte
[3] PLUMgrid Inc.
Sunnyvale, CA
ayesha.ashfaq@seecs.edu.pk; {mali12,ealshaer}@uncc.edu; khayam@gmail.com

ABSTRACT

The inherent design of anomaly detection systems (ADSs) make them highly susceptible to evasion attacks and hence their wide-spread commercial deployment has not been witnessed. There are two main reasons for this: 1) ADSs incur high false positives; 2) Are highly susceptible to evasion attacks (false negatives). While efforts have been made to minimize false positives, evasion is still an open problem. We argue that ADSs design is inherently flawed since it relies on the ADS's detection logic and feature space which is trivial to estimate. In information security e.g. cryptographic algorithms (such as DES), security is inherently dependent upon the key and not the algorithm, which makes these systems very robust by rendering evasion computationally infeasible. We believe there is a need to redesign the anomaly detection systems similar to cryptographic systems. We propose to randomize the feature space of an ADS such that it acts as a cryptographic key for the ADS and hence this randomized feature space is used by the ADS logic for detection of anomalies. This would make the evasion of the ADS computationally infeasible for the attacker.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and Protection*; D.4.6 [**Security and Protection**]: Invasive Software; K.6.5 [**Security and Protection**]: Unauthorized access

Keywords

Evasion; Intrusion Detection Systems

1. INTRODUCTION

While the original models for anomaly detection system were proposed more than two decades ago, anomaly detection still remains an active area of research as the attacks continue to adapt and evade intrusion detection solutions. Current anomaly detection systems employ feature classes for detection that do not vary throughout the operational life of the ADS. Therefore, if an attacker can estimate the normal behavior of the network, it can easily evade the detection system and send attack packets into the network without detection. Moreover, the possibility of detecting such evasion attacks is bleak because the normal network behavior is not frequently updated.

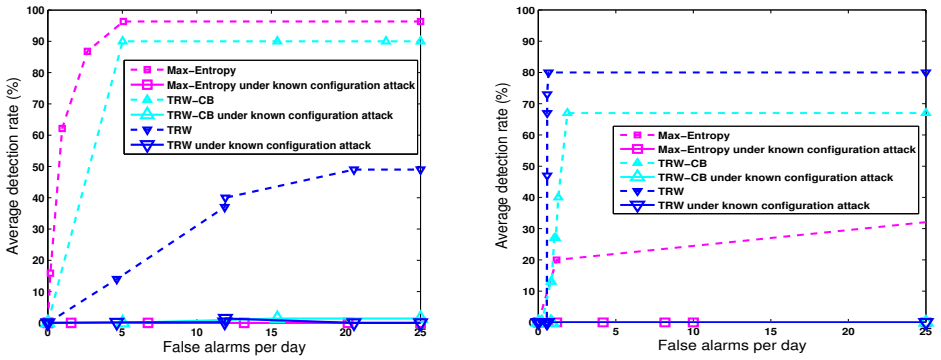
We propose that an anomaly detection system should be redesigned based on mutating feature space i.e. feature space that changes across time. This makes the task of estimating the normal behavior of the network infeasible. Therefore, the attacker would now first have to identify the features being used for detection and then analyze the network traffic pertaining to those features to develop a model of normality. For the attacker to launch the evasion attack, the features used by the ADS for detection should not vary from those estimated. However, if the features change, the attack would be detected because the normal network behavior is now defined in terms of a different set of features than those estimated. Thus, the mutation across time renders the two-tier estimation unrewarding for the attacker since he would have to re-estimate the parameters in different time window. Hence the robustness of the system lies in mutating the feature space across time rather than in the methodology used for detection, just like crypto-algorithms rely on the key for security. Below we discuss the ADS parameters that need to be estimated by the attacker, followed by a case study.

2. CAN WE ESTIMATE THE EVASION MARGIN?

Currently ADSs assume that the underlying detection principle is not known to the attacker. However, it does not hold in real world where some knowledge about the ADS principle can be obtained through methods like social engineering, fingerprinting, etc. [1]. In fact, ADS can be evaded without knowing the exact design principles; several types of attacks (e.g., polymorphic blending attacks, mimicry attacks, etc.) have been proposed in existing literature.

Moreover, ADSs rely significantly on the belief that the attacker does not know the network topology and/or the services running within the network. Hence, it tends to commu-

CCS'13, November 4–8, 2013, Berlin, Germany.
ACM 978-1-4503-2477-9/13/11.
http://dx.doi.org/10.1145/2508859.2512529.



(a) Dataset-1 (b) Dataset-2

Figure 1: Comparison of ADS performance with and without evasion attack using configuration estimation.

nicate with hosts that do not exist or hosts that do not have the requested service available. Thus the malicious packets from the attacker would considerably alter the real-time traffic characteristics from the baseline distribution. However, we show that this is a flawed assumption.

For the attacker to estimate the evasion margin i.e., the number of attack packets that it can send into the network without detection, it has to estimate three ADS parameters: a) baseline distribution; b) realtime distribution; and c)detection threshold.

As a proof-of-concept, we use two existing, prominent and diverse statistical ADSs: Maximum Entropy [2] and TRW/Credit-based TRW [3], [4]. These ADSs are briefly described in Section 3. We now explain the estimation of the ADS parameters below.

The baseline distribution can be estimated by either: a) Observing traffic generated from a host within the target network; or b) Brute force estimation. For the *realtime observation*, a realistic scenario for estimating the baseline distribution is that the attacker compromises a host X in network A that communicates with the target host [1]. Hence, it can observe the normal traffic from host X to the target entity that can be used to build the normal profile for network A. *Brute force estimation* is the most common modus operandi used extensively in cryptanalysis. Despite its computational complexity, it has been shown that with the current high-performance COTS (multithreaded and multicore) hardware, it is not difficult for a craft attacker to acquire and exploit hardware parallelism to carry out a bruteforce analysis [1], [5].

We proposed a Markovian stochastic model [6] of temporal dependence in an ADS's anomaly scores. While the motivation for the original work was to improve the accuracy and automation of an ADS using threshold estimation, the technique can be adapted to estimate the ADS detection threshold for evasion. However, we restrict the evaluation of conditional entropy to first order Markov chains only:

$$H\left(p_j|p_i\right) = -\sum_{\omega} p(p_i, p_j) \log\left(p\left(p_j|p_i\right)\right). \quad (1)$$

The conditional entropy $H(p_j|p_i)$, of two random variables p_i and p_j correspond to the information in p_j not given by p_i. Thus, computing the maximum conditional entropy between baseline distributions in two consecutive time bins, as we slide from bin 1 to bin n, can provide us the minimum information overlap in normal benign data. This can be stochastically modeled as:

$$H_{\max} = \max_{i,j\in\{1,2\ldots n\}} H\left(p_j|p_i\right) \quad (2)$$

This minimum information overlap can be used to identify the acceptable divergence bounds for normal traffic for which the detector does not raise an alarm.

Once the baseline distribution and the threshold have been estimated, the detection logic can be used to estimate the realtime distribution. This realtime distribution can then be used by the attacker to generate attack sessions in different feature classes that stay below the threshold and hence evade ADS detection. For example, for the Maximum entropy ADS, it can be estimated as:

$$\widehat{q_B}[\omega] < 2^{\frac{\tau_{KL}}{p(\omega)}} p(\omega) \quad (3)$$

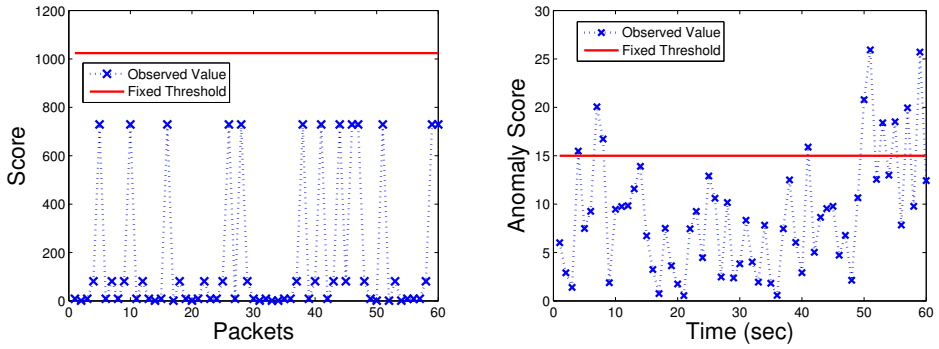
where $q(\omega)$ is the realtime distribution, τ_{KL} is the detection threshold and ω is the packet class

3. CASE STUDY

The Maximum Entropy detector [2] employs Kullback-Leibler (KL) divergence measure for anomaly detection. The measure computes how much the baseline distribution $p(\omega)$ varies from the real-time distribution $q(\omega)$. Traffic is divided into 2348 packet classes based on the destination ports and the protocol. The detector uses maximum entropy estimation to develop the baseline distribution for the traffic classes. If the divergence between the baseline and the real-time distributions for a particular packet class exceeds the threshold τ_{KL} in h of these W windows, an anomaly is flagged by the detector. Maximum entropy has been shown to provide high accuracy dividends [7].

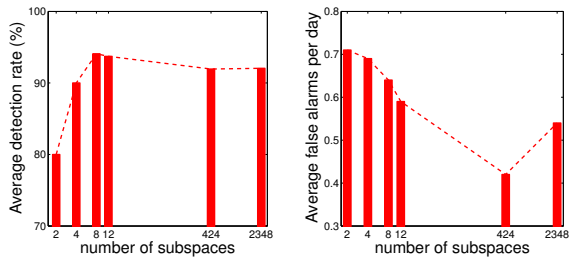
Sequential hypothesis testing based TRW ADSs [3] [4] employ the likelihood ratio test to identify if local/remote hosts are scanners. TRW classifier [3] detects remote scanners while Credit-based TRW (TRW-CB) [4] detects local scanners. Both these algorithms have been shown to be quite accurate and commercial ADSs also deploy these algorithms for portscan detection.

For both Maximum entropy and TRW ADSs, we launched stealthy scanning attack by estimating the ADS's parameters as described in Section 2. Both the detectors failed



(a) TRW (b) Maximum Entropy

Figure 2: Threshold values observed in stealthy scanning time window for TRW and Maximum Entropy



(a) Detection rate (b) False alarms per day

Figure 3: Accuracy of the Maximum Entropy detector with varying number of feature classes.

to detect the scanning probes on two different datasets as shown in Figure 1.

Figure 2(a) shows that TRW failed to detect the scan traffic in a malicious time window. It can be clearly observed that the likelihood ratio did not exceed the threshold. TRW calculates the likelihood ratio for each connection attempt and classify them as anomalous if the likelihood ratio for any host increases more than the threshold. Since the hosts were also generating benign traffic, the likelihood ratio did not cross the threshold set. Similarly, Figure 2(b) shows the threshold values observed in a 60 sec time window for Maximum entropy. As proposed in [2], if the threshold (15) is exceeded 30 times within 60 seconds interval, an alarm is raised. Due to stealthy scanning the run-time distribution exceeded the threshold only 10 times. Therefore, just by slowing down or the effect of averaging out in the normal traffic, scanning was able to go undetected.

It can be observed that of the three parameters that an attacker has to estimate (Section2), two of those are dependent on the features employed by the ADS for detection. Since these features are inherently fixed by design, the attacker can easily estimate them to evade the ADSs.

4. DOES A LARGER FEATURE SPACE GUARANTEE OPTIMAL PERFORMANCE?

Using a high-dimensional feature space does not necessarily yield optimal performance. Figure 3 provides the accuracy gain on the endpoint dataset achieved by the Maximum Entropy ADS as the feature space is varied from 2 to 2348 using the detector's slicing technique progressively. Figure 3(a) shows that the detection rate does not increase proportionally as we increase the number of analyzed static feature space from 2 to 2348. The same trend is observed in the false alarms as well. Thus, using all the features simultaneously does not ensure high accuracy. Hence selecting a few feature class(es) judiciously for detection in a time window would suffice to introduce enough randomness in the process of ADS detection, so as to render the parameter estimation attacks impractical.

5. MOVING TARGET-BASED ADS DESIGN

Since current ADSs are susceptible to evasion margin estimation, they can be bypassed by an intelligent and a resourceful attacker. This is because the underlying feature space does not change with time and hence can be leveraged to analyze network traffic semantics and craft attacks accordingly. We believe a moving target based ADS design would, however, make it infeasible for an adversary to craft evasion estimation attacks due to inherent randomness introduced by the mutating feature space. However, it is important to identify the right number of features to be employed for detection in different time windows, for optimal performance.

6. REFERENCES

[1] D. Wagner and P. Soto, "Mimicry attacks on host-based intrusion detection systems," ACM CCS 2002.

[2] Y. Gu, A. McCullum and D. Towsley, "Detecting anomalies in network traffic using maximum entropy estimation," ACM/Usenix IMC, 2005.

[3] J. Jung, V. Paxson, A. W. Berger, and H. Balakrishnan, "Fast portscan detection using sequential hypothesis testing," IEEE Symp Sec and Priv, 2004.

[4] S. E. Schechter, J. Jung, and A. W. Berger, "Fast detection of scanning worm infections," RAID 2004.

[5] C. Smith, A. Matrawy, S. Chow and B. Abdelaziz, "Computer worms: Architectures, evasion strategies, and detection mechanisms," Journal of Information Assurance and Security, 2008.

[6] M.Q. Ali, H. Khan, A. Sajjad and S.A. Khayam, "On Achieving Good Operating Points on an ROC Plane using Stochastic Anomaly Score Prediction," ACM CCS, 2009.

[7] A.B. Ashfaq, M.J. Joseph, A. Mumtaz, M.Q. Ali, A. Sajjad and S.A. Khayam, "A comparative evaluation of anomaly detectors under portscan attacks," RAID 2008.