International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 3, Issue 6, June 2016)

Key-Recovery Attacks on KIDS, A Keyed Anomaly Detection System

Mahananda
P G Student
Department of Computer Science & Engineering
PDACE, Kalaburgi, Karnataka, India
maanuhannur@gmail.com

Chandrakanth Biradar
Professor
Department of Computer Science & Engineering
PDACE, Kalaburgi, Karnataka, India

***Abstract* –With the anomaly detection systems, many approaches and techniques have been developed to track novel attacks on the systems. Anomaly detection systems based on predefine rules and algorithms, it is difficult to define all rules, To overcome this problem various machine learning schemes have been introduced, In this schema, the system relies on deriving models of normality that is later used to detect suspicious events, Such algorithms are generally susceptible to deception, notably in the form of attacks carefully constructed to evade detection. Various learning. Schemes have been proposed to overcome this weakness. One such system is keyed IDS (KIDS), the KIDS core idea is akin to the functioning of some cryptographic primitives, namely to introduce a secret element (the key) into the scheme so that some operations are infeasible without knowing it. In KIDS the learned model and the computation of the anomaly score are both key dependent, a fact which presumably prevents an attacker from creating evasion attacks KIDS depend on secraecy of the key and the method used to generate the key each time when attacker attacks.**

1. INTRODUCTION

A wireless sensor network (WSN) consists Intrusion detection is an important defense mechanism used by defenders to determine if someone has penetrated their system. Two approaches have typically been taken when designing intrusion detection systems: signature-based and anomaly detection. Signature-based systems, such as Snort, match incoming packets against various signatures that represent different types of malicious activity, such as particular buffer overflow attacks or signatures for worms. Unfortunately, such a system is reactive in that a malicious activity must first exist before a signature can be developed. Anomaly detection attempts to address this shortcoming by alerting on changes in activity, where these changes are unusual (anomalous). A great deal of research effort has gone into creating anomaly detection systems[6], although very few systems have seen wide-
Spread use. Such systems have been developed to operate at the host level to detect if a user is attempting to abuse an application in order to gain root privileges (e.g., Forrest et al. [3]), and at the network level to detect if a remote adversary is attempting to gain unauthorized access (e.g. Minds [1]). However, little work has gone into determine- in if the underlying assumptions hold. In particular, it is assumed that the malicious behavior is anomalous, and therefore that by detecting anomalous behavior we are detecting malicious behavior.

Recent work has accurately pointed out that security problems differ from other application domains of machine learning in, at least, one fundamental feature: the presence of an adversary who can strategically play against the algorithm to accomplish his goals. Thus, for example, one major objective for the attacker is to avoid detection. Evasion attacks exploit weaknesses in the underlying classifiers, which are often unable to identify a malicious sample that has been conveniently modified so as to look normal. Examples of such attacks abound. For instance, spammers regularly obfuscate their emails in various ways to avoid detection, e.g., by modifying words that are usually found in spam, or by including a large number of words that do not Similarly, malware and other pieces of attack code can be carefully adapted so as to evade intrusion detection systems (IDS) without compromising the functionality of the attack.
A few detection schemes proposed over the last few years have attempted to incorporate defenses against evasion attacks. One such system is a keyed intrusion detection system (KIDS), introduced by Mrdovic and Drazenovic at DIMVA[1]. KIDS is an application-layer network anomaly detection system that extracts a number of features ("words") from each payload. The system then builds a model of normality based both on the frequency of observed
Features and their relative positions in the payload. KIDS' core idea to impede evasion attacks is to incorporate the notion of a "key", this being a secret element used to determine how classification features are extracted from the payload. The security argument here is simple: even though the learning and testing algorithms are public, an adversary who is not in possession of the key will not know exactly how a request will be processed and, consequently, will not be able to design attacks that thwart detection.

KIDS is an application-layer network anomaly detection system that extracts a number of features ("words") from each payload. The system then builds a model of normality based both on the frequency of observed features and their relative positions in the payload. KIDS' core idea to impede evasion attacks is to incorporate the notion of a "key", this being a secret element used to determine how classification features are extracted from the payload. The security argument here is simple: even though the learning and testing algorithms are public, an adversary who is not in possession of the key will not know exactly how a request will be processed and, consequently, will not be able to design attacks that thwart detection

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 3, Issue 6, June 2016)

The Proposed work is organized in 6 sections. Section 1 presents a general introduction of a network intrusion detection system using secret element key. Section 2 presents the related work of the different types of security methods. Section 3 presents the design of the proposed system with block diagram are discussed. Section 4 presents Results and Discussion.Section5concludes the work with future enhancement.

2. Related Work

detect such optimally modified instances by adapting thedecision surface of the classifier, and also discuss how the adversary might react to this.

The authors [2] considered a problem, i.e. an adversary with full knowledge of the classifier to be evaded.

The author [3] considered a problem, i.e how evasion can be done when such information is unavailable. They formulate the adversarial classifier reverse engineering problem (ACRE)[2] as the task of learning sufficient information about a classifier to construct attacks, instead looking for optimal strategies. The authors use a membership oracle as an implicit adversarial model: the attacker gives the opportunity to query the classifier with any chosen instance to determine whether it is labeled as malicious or not. Consequently, a reasonable objective is to find instances that evade detection with an affordable number of queries. A classifier is said to be ACRE learnable if there exists an algorithm that finds a minimal-cost instance evading detection using only polynomial-many queries.

The authors [4], Consider the problem a classifier is ACRE k-learnable if the cost is not minimal but bounded by k. Among the results given by the author 3, it is proved that linear classifiers with continuous features are ACRE k-learnable under linear cost functions. Therefore, these classifiers should not be used in adversarial environments.

The authors [5], demonstrate that polymorphic mimicry worms, based on encryption and data encoding to obfuscate their content, are able to evade frequency distribution-based Anomaly detectors like PAYL. PAYL models byte-value Frequency distributions (i.e., 1-grams), so detection can be avoided by padding anomalous sequences with an appropriate amount of normal traffic. In order to counteract polymorphic mimicry worms, PAYL authors developed Anagram [7], an anomaly detector that models n-grams observed in normal traffic.

From the literature survey, we found that such algorithms are generally susceptible to deception,notably in the form of attacks carefully constructed to evade detection.overcomes this weakness we propose KIDS-keyed intrusion detection system.

The proposed method is based on standard cryptographic primitives included secret element (the key), some operations is infeasible without knowing it.

3. Proposed System

The communication between source and end user, in the key server key will be generated each time when communication done.

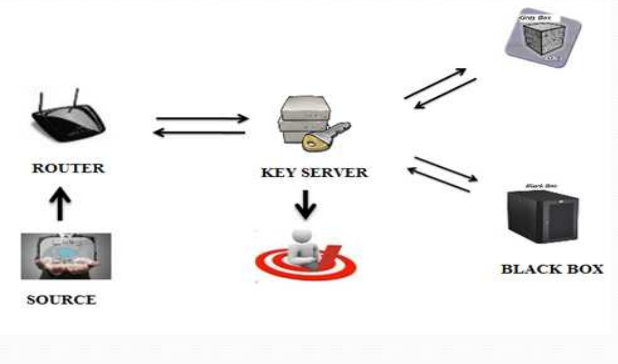


Fig 1. Key generating in key server
Fig 1 shows key generating in key server.

.

In this section various attacks on kids aimed at recovering the secret set of delimiters(i.e, the key).for these attacks two broad classes,depending on what feedback from KIDS the attacker may have access to before presenting our attacks. Getting feedback from the IDS seems a priori more problematic, but it would be unsafe to assume that this knowledge is unavailable to the attacker. In the case of the black box model, one potential scenario involves an attacker who can determine whether an alarm has been generated or not. This information could be obtained by observing the network and checking if an alarm is sent to the security officer,either directly by observing the channel or indirectly through some side channels. If the attacker is an insider, even one with fewer privileges, obtaining this information may be easier.

The gray-box model is stronger in the sense that getting access to the anomaly score seems rather unrealistic. Apart from the merely theoretical interest, sender believe that the score may be also obtained by the attacker if, for example, such a value is included in the alarm sent to the security officer. Some real-world IDS do this in order to provide the decision maker with as much information as possible about the potential attack. Thus, if such alarms are not encrypted, an observer could get access to the score.

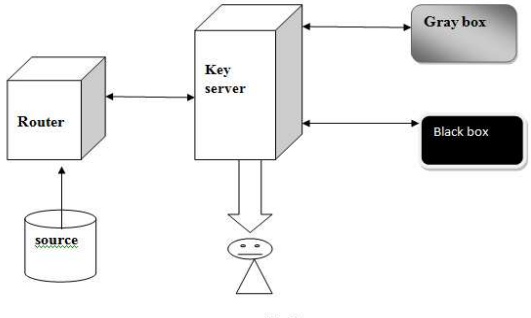


Fig 2. Block diagram of keyed intrusion detection system

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 3, Issue 6, June 2016)

The fig 2 shows the key detection.

Source-In this module, client browses a file, encrypt and upload to the router. Generate the key Request to get the key for the file.

Key sensor-Matches a key for new file with gray box and black box. For the new file key will be stored in both black box and gray box, If key already exists mean it inform to use the same key which is already available, Check key's Safe (attacked or not) and capture all attackers, Finding all end user requested file keys.

Router-Receive Enc data from source, Get Key from Gray Box or Black Box to download the file. Decrypt data when end user request, Send file to end user, View all files transaction

Receiver-Request secret key and available files in the router. Request and receive decrypted files.

A. KEY Recovery attacks

Author Juan E. Tapiador, Agustin Orfila, Arturo Ribagorda, and Benjamin Ramos[9] experiment analysis shows that in KIDS scheme attacker easily able to interact with it and using the feedback of the interaction attacker attacks on the secure data. Attacker takes help of various queries to get more information related to the secret key. The attack makes exactly 257 queries to KIDS: 256 with each tentative key element d, plus one final query to determine which subset corresponds to the key [8].

B. Adversarial Model and Notation

When assessing the security of systems such as KIDS, one major problem comes from the absence of widely accepted adversarial models giving a precise description of the attacker's goals and his capabilities. Barreno et al. [4] Have recently introduced one such model for secure machine learning[5]and discussed various general attack categories.

Our work does not fit well within Barreno et al.'s model because our main goal is not to attack the learning algorithm itself, but to recover one piece of secret information that, subsequently, may be essential to successfully launch an evasion attack. In some respects, our work is far more similar to that of Lowd and Meek [1], where the focus is on the role of active experimentation with a classifier. In such a scenario, it is absolutely essential for the attacker to be able to: (1) send queries to the classifier; and (2) get some feedback about the properties of the query as processed by the system. We emphasize that the ability to do this is close to the bare minimum required to analyze the security of any scheme.

C. Key-Recovery on Gray-Box KIDS

In this attack, we assume the attacker has access to the anomaly score assigned to a chosen payload. Furthermore, it is reasonable to assume that some normal payloads are known for. (Consider, for example, the case of an IDS analyzing HTTP requests sent to a publicly accessible web server, where a large number of such payloads will be known by the attacker.)

Let p be one such normal payload. A straightforward strategy to identify what elements of p belong to the key D consists of feeding KIDS with the first byte of p, then with the first two bytes of p, and so on. When the next to- the-last byte happens to be a delimiter, the KIDS will detect a transition where the left word is likely to have been seen during training, whereas the right word is often unknown (since it is truncated). At this point, the anomaly score will suffer a slight decrement. By conveniently repeating them procedure, all the delimiters present in p can be recovered.

Regardless of the technical details, the main drawback of the naıve strategy discussed above is that the attacker will only be able to recover those key elements present in the normal payloads available, which may well be just a fraction of all of them. Besides, the complexity of such an attack is linear in the number of payloads and their lengths.

D. Key-Recovery on Black-Box KIDS

In this section we present a key-recovery attack when the Only information about a payload an adversary gets from KIDS is its classification label, i.e., whether it is normal or Anomalous. In some respects, this information is less fine grained Than the anomaly score, so it is reasonable to expect That attack working under this assumption will be slightly More complex.

The central idea behind our attack is actually quite simple. We will provide KIDS with a normal payload concatenated with a carefully constructed tale. Such a tail contains a large number of unseen words separated by the candidate delimiter. If the delimiter does not belong to the key, the entire trail will be processed as just one.

The word and the anomaly score will be roughly similar to that of the original payload. If this is the case, then the payload will be marked as normal with high probability. Conversely, if the delimiter does belong to the key, the tail will be fragmented into a large number of previously unseen words and transitions. This will negatively impact the anomaly score, invariably resulting in an anomalous payload.

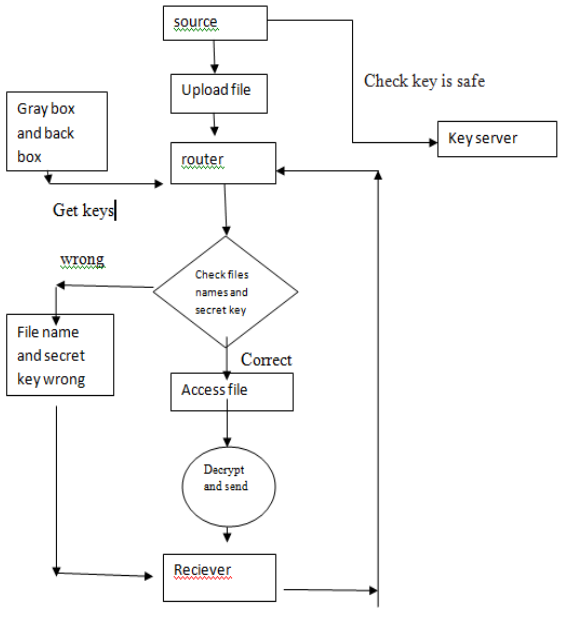


Fig 5. Flow chart of KIDS

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 3, Issue 6, June 2016)

4. Result Analysis

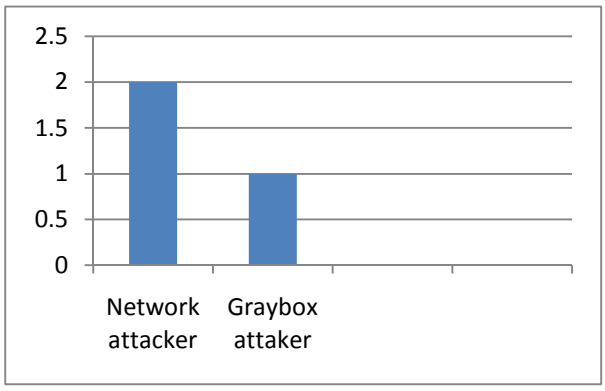
We experimentally validate our attacks with an implementation of KIDS written in C. The system was trained with 2000 HTTP payloads captured in a university network. The data set does not include attacks, as they are not necessary to recover the key.

Following the design principles given in [9], our experiments have been conducted with key sizes ranging from 15 to 30, even though this parameter has little influence on the results. In all cases, the delimiters are randomly generated avoiding repetitions, and the detection threshold is chosen to guarantee that at least 99 percent of the training set falls below it.

We note that this way of selecting a key does not coincide with the procedure given in [9], where the authors suggest a method involving both normal and attack traffic. This, however, is irrelevant to our attacks, as they worked on an already trained system, regardless of how the key has been chosen.

In the case of the gray-box attacks, words and are automatically extracted from one normal payload .

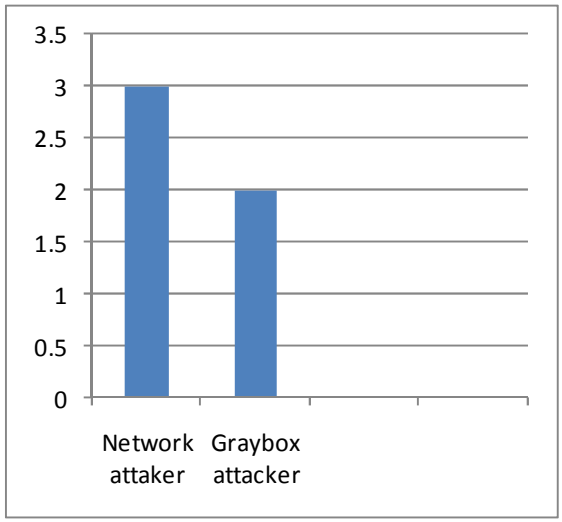
Since the black-box attacks, we used a subset of randomly Chosen payloads and made them available to the attacker.



Attack type

Number of attacks, Network attacker =2
Number of attacks, Gray box attacker=1

Fig 6.Number of attacks



Attack type

Number of attacks, Network attacker= 3
Number of attacks, Gray box attacker=2

Fig 7. After attacks

5. Conclusion

We have analyzed the strength of KIDS against key-recovery attacks. We have presented key-recovery attacks, according to adversarial settings, depending on the feedback given by KIDS to probing queries. Analysis showing that it is reasonably easy for an attacker to recover the key. Our focus of this work has been on recovering the key through efficient procedures, demonstrating that the classification process leaks information about it that can be leveraged by an attacker. However, the ultimate goal is to evade the system, and we have just assumed that knowing the key is essential to craft an attack that evades detection or, at least, that significantly facilitates the process. It remains to be seen whether a keyed classifier such as KIDS can be just evaded without explicitly recovering the key.

References

[1] D. Lowd and C. Meek, “Adversarial Learning,” Proc. 11th ACM SIGKDD Int’l Conf. Knowledge Discovery in Data Mining (KDD ’05), pp. 641-647, 2005.
[2] Biggio, G. Fumera, and F. Roli, “Adversarial Pattern Classification Using Multiple Classifiers and Randomisation,” Proc. IAPRInt’l Workshop Structural, Syntactic, and Statistical Pattern Recognition, pp. 500-509, 2008.
[3] Metasploit Framework, www.metasploit.com, 2013.
[4] M. Barreno, B. Nelson, A.D. Joseph, and J.D. Tygar, “The Security of Machine Learning,” Machine Learning, vol. 81, no. 2, pp. 121-148, 2010
[5] M. Barreno, B. Nelson, R. Sears, A.D. Joseph, and J.D. Tygar, “Can Machine Learning be Secure?” Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS ’06), pp. 16-25, 2006.
[6] C. Gates and C. Taylo, “Challenging the Anomaly Detection Paradigm: A Provocative Discussion,” Proc. New Security Paradigms Workshop (NSPW), pp. 21-29, 2006.
[7] K. Wang, J. Parekh, and S. Stolfo, “Anagram: A Content Anomaly Detector Resistant to Mimicry Attack,” Proc. Ninth Int’l Conf.

Recent Advances in Intrusion Detection (RAID '06), pp. 226-248, 2006.

[8] Juan E. Tapiador, Agustin Orfila, Arturo Ribagorda, and Benjamin Ramos "Key-Recovery Attacks on

[9] KIDS, a Keyed Anomaly Detection System" IEEE transaction on DEPENDABLE AND SECURE COMPUTING, VOL. 12, NO. 3, MAY/JUNE 2015

[10] B. Nelson, B.I.P. Rubinstein, L. Huang, A.D. Joseph, and J.D. Tygar, "Classifier Evasion: Models and Open Problems," Proc. Int'l ECML/PKDD Conf. Privacy and Security Issues in Data Mining and Machine Learning (PSDML '10), pp. 92-98, 2011