Implementation of Insider Threat Detection System Using Honeypot Based Sensors and Threat Analytics

Muhammad Mudassar Yamin[1], Basel Katt, Kashif Sattar[2], Maaz Bin Ahmad[3]

[1] Norwegian University of Science and Technology, Gjovik, Norway
[2] University of Arid Agriculture Rawalpindi, Pakistan
[3] PAF KIET Karachi, Pakistan
muhammad.m.yamin@ntnu.no,basel.katt@ntnu.no,kashif@uaar.edu.pk, Maaz@pafkiet.edu.pk

Abstract

An organization is a combination of vision, technology and employees. The wellbeing of organization is directly associated with the honesty of its workers. However, an organization is also threatened by misuse of information from its agents like former employees, current employees, vendors or business associates. These kinds of threats which are posed from within the organization are known as Insider Threats. Many approaches have been employed to detect the Insider Threats in organizations. One of such approaches is to monitor the system functions to detect possible insiders. These approaches raise unnecessary amount of false positive alarm which is then taken care of with the use of evolutionary algorithms. The solution to this Insider Threat detection requires a lot of configuration before implementation in real world scenarios due to different threshold values in different organizations. Insider Threat detection can be done by means of honeypots sensors in a limited and in satisfactory way. The present research proposes a new technique for detecting insiders using encrypted honeypots. This technique complements the existing insider detection systems and improves its performance in terms of decreasing false positive results.

Keywords: Insider Threat, System monitoring, Activity detection, Honeypots, Threat analytics

1. Introduction

In this section we discuss about the background study of Insider Threat. Kevin Mitnik once stated, "Companies spend millions of dollars on firewalls, encryption and secure access devices, and its money wasted, because none of these measures address the weakest link in the security chain". That quote completely gives a good scenario of tension faced by network security professionals in the modern days. The weakest links in the security chain are the individuals who work in secure networks with authorization. They can produce a danger to company if they intentionally or unintentionally damage networks infrastructure or steal important information. As described by MALLAH [7] these kinds of individuals are called insider. Both type of the attacker can damage the vulnerabilities of the system. Therefore, it is very important to implement an efficient and effective security policy for the removal of such threats.

According to Moore [14] the authorized users or employees have access to the confidential data and to the sensitive assets of the company, so there is always a risk that the employees may misuse this data access for any mischievous purpose. An example of insider case is Chelsea Manning who was responsible for the leaking more than 60000 U.S department of defense documents on WikiLeaks and Edward Snowden who exposed secret NSA documents in public. These two cases are important examples of Insider Threat incidents. In these cases, detection of insiders was very important because of the data breach in the National Security issues of Unite States and people lives were also endangered. the frequency of Inside attacks during the same years were 29, 20 and 32 percent respectively.

The Computer Emergency Response Team (CERT) in USA actively researches on Insider Threat and release surveys from time to time regarding Insider Threat detection. The recent surveys published in 2006 by CERT/CC's produced following findings regarding inside and outside threats. The findings revealed that the number of Outside attacks during year 2004, 2005 and 2006 were 71, 80 and 68 percent respectively.

Early research about the Insider Threat is conducted by Hayden [9] which suggested that it is the need of the hour that the continuous check on the employee activities must be maintained in the company to avoid any discrepancy. The continuous profiling of the user activities help to locate any Insider Threat well before time and the administration can the easily avoid, detect and recover this Insider Threat timely. The company should maintain a check on the detailed behavior of its employees to detect the threatening activities of the au-theorized users. The company should study the profile of the trusted users in the light of the organization's policy. After the investigation, if any user is found to be involved in any suspicious activity which is against the policies of the organization, then the user who possesses the suspicious profile is marked as suspicious and onward the more careful watch should be maintained over that user.

The problem is that the current techniques of detecting Insider Threat got many imitations. The limitations are related to the profile checking of the employees because the Insider Threat detection technique developed by researchers[1], can give rise to many false positive alarms during the detection process, the production of false positive alarms is responsible for the unnecessary system lock down of the company. The technique of detecting the Insider Threat gives rise to false positive alarms for the company and the frequency of these false alarms can be cut down by eliminating the human error factor. The human error is thus responsible for the large amount of false positive alarms in the company. It can be explained as follow. The employee checking the main server of the company may get confuse about the action of another employee on the system and he could assume the simple file access log as a bad intention of the employee for the company. He may raise a false positive alarm of Insider Threat detection against that employee. He may not be necessarily right every time. Therefore, it is very important that a mechanism should be developed which would be very accurate in the raising of real Insider Threat detection alarm and detect only the genuine, real and imminent threat and not raise any false positive alarm.

1.1. Insider Threat

Any company or organization has the resources to locate or control the situation when an outsider (non-representative) tries to steal the organization data physically or electronically. Their rivals constantly threaten the organization that they might steal their data illegally. In most of these cases, when a large amount of sensitive data is stolen form the company, then it is very difficult to find out the culprit because in the majority of cases, the criminal lies within the victim organization. The culprit is most of the times the insider- a specialist who has easy access to all the company's data. That insider could be dangerous because he has access to all the company's assets and secrets. That insider may do this crime for his own satisfaction or he may also be a "spy" of any rival company doing this task to get monetary benefits. The insider may steal the company's data and items to sale this information to another enterprise to strengthen their credibility.

In majority of companies, the employees are given accounts to get easy access to company's data. In case of Insider Threat detection, these accounts are very useful to detect any data breach in the company system. The Insiders (infiltrated employees) got their accounts in the respective company, which allows them to use the PC frameworks of the company without any problem.

Insiders made use of the accounts given to them by the company which allow them legally to use the PC frameworks of the company. In the previous years, this permission allowed the insiders to conduct the execution of these commitments; these permissions could be misused by insiders to harm the business of the company. Insiders have a capacity to get knowledge of the company's assets and they also possess licensed innovation of the systems that are designed to secure company's data. So these access codes make it less challenging and easier for the insider to roam through any security control check which they know. The concrete proximity to the company's data reflects that the insider doesn't does not need to hack into the hierarchical system through the outside border by navigating firewalls; like which is done in case of buildings, mostly with quick access to the association's inside system. Insider Threats are very difficult to execute, and it needs a lot of efforts on behalf of insider, this statement is given on the basis that the insider has an authorized access to the company's data and resources. The main purpose of insider is to get access to the company's data without the knowledge of the relevant authorities. The insiders then misuse this information to gain their own benefits.

The damage which is caused by this Insider Threat can be of various types. This can be understood with the help of examples of burglary, robbery or extortion. The insider breach in the company's data by introducing a virus, worm or Trojan in the system is similar to the burglary of money from bank. Like the money is stolen from bank with the effort of robber, same is the case of company's data which can be stolen from the data center by the effort of insider. The danger of Insider Threat can be encountered by taking appropriate security measures. These security measures include implementation of ethical policy for Internet surfers, use of different spy product examining programs, hostile to infection projects, firewalls, and a strong information security check and chronicling administration.

1.2. The Insider

There are a number of different associations like CERT and RAND which perform analysis of Insider Threat and the issues related to the Insider Threat detection [10]. The significance of the term insider has to be dealt in detail for a clear picture now. Insiders are a group of people in a particular company which are involved in any illegal activity in the company and are mostly followed by digital security group and especially by Insider Threat researchers. Many of the researchers suggest a duel way of dealing with the identification of Insider Threat, recommending that the insider may come under the mark of a quantifiable parameter, for example, and an occupation classification. The assumptions which is based on this methodology in this case, resulting out to be somewhat vague and doubtful with the expanding utilization of outsourcing, versatile figuring, contractual workers, and business associations. In the present research; the researchers have made use of a grid approach to describe and characterize the insider in light of access (Bishop, M., 2005). The researchers have nominated and described an insider as a trusted entity that has given the potential to abuse a security center. The insider is detected and resolved in relation to a set up security arrangement. Insider Threat lies in the entrance and capacity to:

1. Violate a security approach utilizing honest to goodness access, or
2. Violate an entrance control approach by getting unapproved access

1.3. The Insider Problem

The complexities which are associated with Insider Threat detection are more complex than the written theory. It is much more than management of outer digital threat with reference to objective demographic. The problem arises mostly because of the matters of managing trust, security, and morals, which the researcher investigates later. The Insider Problem can be summed up as the test of protecting an association from interior digital attacks. The problem is that most of the companies do not give this matter due importance and take it for granted. The companies think that the issue of Insider Threat detection is easy to handle but they don't know that this matter can prove to be really dangerous for their company in future.

1.4. Ethics and Trust

One of the important challenges about the insider issue is trust and confidentiality. Whenever any person is doubted to be an insider, there is first some level of trust associated with that specific person. The provision of this trust means that the respective insider is a trusted employee of the company and got access to PC account, access to a confined room and surely possesses access to a framework asset and delicate data of the company as well. The point to be noted over here is that when an individual is allowed such type of trust in a company, then that person automatically gets the authority to misuse this trust for his own benefits.

1.5. Cyber Attacks

There are three forms by which digital attacks could be performed. These forms are: exfiltration, information defilement, and refusal of administration. These digital attacks can cause a huge amount of destruction to the company. The damages may include harmed believability, uncovered helplessness, and budgetary misfortunes. In order to prevent such digital attacks, first it is very to understand what is a digital attack, how it works and what are the different types of digital attack. After getting knowledge about the digital attack, the investigator can then have a look at real life insider attack cases and get to know that how the attack occurred and what happened on the utilized frameworks utilized during the attack.

1.6. Exfiltration

An exfiltration attack consists of information robbery, IP burglary, or any intentional expulsion of information. This attack is the result of grouping up of insiders with the people of enemy contender groups of the company [6]. The insiders usually hand over touchy or private data to these enemies for financial benefits.

The most infamous and dreadful example of exfiltration in the history is that of a professional FBI operator Robert Hanssen who worked in the secret services of USA. Hanssen worked in the FBI as an agent but actually he was a spy of Soviet Union. He was an insider in FBI. Working in the disguise of FBI agent, Hanssen downloaded a large amount of classified data to encoded information by setting up the equipment and a portable workstation. He further used the portable PC to speak with Soviet insight officers for the exchange of the data that he got. This attack led the Hanssen expose a large amount of extraordinary classified information of USA to Soviet Union. With a specific end goal of acquiring access the data, Hansen got to the FBI computerized records framework. Hanssen hacked the data center of FBI and extracted all the required information. Hanssen constantly put demand the information from the system which was not of his concern. He was an approved client of the database of FBI, therefore his intentions were never doubted. Therefore, Hanssen conducted the attack on the data of organization without raising much suspicion. The conclusion is if FBI has put ample check on the suspicious activities of its employees, then this data breach is not possible to occur.

Another similar case of exfiltration happened in CIA. The culprit was a 16 years old employee of CIA Harold Nicholson [10]. He was employed in CIA as a teacher in a CIA exceptional preparing focus from 1994 to 1996. During his stay at the CIA, Harold transferred a huge amount of classified data to Russia. Harold get hold of the data by hacking into the CIA's venture PC framework and by conducting a vast kind of inquiries on the CIA's databases. These inquiries were out of range of his discretion in the CIA and consisted of information regarding US insight information on Chechnya, which he sold to Russian authorities.

The CIA then took help from the FBI to investigate this matter and Harold also passed through a standard polygraph test successfully. They found out that Harold transported a big number of Top-Secret documents from CIA PCS to Russia onto scrambled plates, movies of film, and also by his PC hard drive. This huge amount of data transfer from the CIA could have raised suspicion if the data traffic on the CIA framework have been continuously monitored. However, due to the absence of any security check, the fact remained unknown that when and how Nicholson hacked the framework of CIA and leaked the classified information.

1.7. Data Corruption

Another type of Insider Threat is that of data corruption. An information defilement attack is described to be as a strange change in the real data or information of the system. It consists of modification and removal of data from the system. This attack is basically meant for digital extortion or in attempt to harm information of the system. When data is changed in a system, then a lot of problems could be produced for the company.

An example of case of data corruption can be studied with reference to programming engineer Chris Harn [10]. Harn was the supervisor of PC system and sever checking at Autotone Systems. The purpose of these servers were to digitalize the daily salary of workers. Harn manipulated his power and authority by changing the data at the framework of Autotone Systems of super-client benefit program and changed the salaries of the worker colleagues for more than 3$ of the illegal profit of the company. By changing the salaries of the workers, Harn changed the data of company review detail to cover up his corruption of money for the increased salaries of the employees. Harn’s access to framework of company allowed him to do this task of data corruption in the system. The bottom line of this discussion is that a huge amount of data corruption was being executed in the database of the company. Eventually, the discrepancy between the wages and salary reviews were balanced in the Autotone’s framework. If any strange activity in the database of company have been checked regularly, then we can avoid such incidents.

1.8. Denial of Service

A ridiculous attempt to make a PC information inaccessible to its planned clients is called administration attack. This attack is carried out by bringing changes in the structures of company data, for example, these changes can be produced by the help of malignant code, malware establishment, over-burdening a support, or by another activity that results in damaging a company’s data .

Some examples of administration attack/assaults however are significant in the sense that they are not particularly acknowledged or executed on the spot. These attacks are characterized by the attacker induced malware which causes all the data infiltration after a certain period of time. The effect of this induced malware may take minutes, days, months, or even years to show full execution. These attacks show their effect when they are "exploded" or triggered by an occasion or a point in time. This is the reason these attacks are called 'bombs'. These "bombs" can appear in two shapes. The first of these is a rationale bomb, which is triggered by a framework occasion. Rationale bombs lay dormant and don't execute the effect until a specific framework occasion happens. Unlike rationale bomb the second type of 'bomb', is called a period bomb, which is triggered by a particular minute in time. These bomb attacks are not activated by an occasion. They lay dormant silently until a timeframe is set by the insider.

1.9. Abnormal Activity

A lot of cases of insider attacks like exfiltration, information debasement and foreswearing of administration assaults happen on daily basis in different companies and cause a lot of damage and monetary losses. The implementation of Insider Threat detection procedure in the company can check the irregular activities of insiders/attackers in the company’s framework. The insider before initiating his attack can take first step by starting the unusual activities on the system. An efficient Insider Threat detector takes notice of these abnormal and unusual activities of the insider and pin point them in the beginning. An example of this unusual conduct is Hansen's unusual hunt of information , Nicholson's extraordinary huge information exchanges, Harn's information and review record alteration, Cooley's late night access and erasure of $2.5 million worth of engineering drawings, and Shae's cancellation of review information. These all examples reflect examples of irregular activities that happened during or before the insider attack. A framework that made regular security checks of the company’s employees accounts could have distinguished these variations from the normal and identified the attackers. This reason along with the above cases and numerous more case of genuine insider attacks are the inspiration for this present research.

The parameters associated with Insider Threat detection is more complex than external cyber threats. The employee trust level and intention make the situation very complicated, so simple profiling of employees can easily be over looked by employee intention in case of an incident. Which make it very difficult for accurate detection of Insider Threat which raises a lot of false alarms. These false alarms also create mistrust among employee and employer. Researchers are able to reduce the amount of false alarms using evolutionary algorithms. But the problem associated with employee intention remains unfixed. To fix the issue researcher proposes a new technique for detecting Insider Threat using encrypted honey pots. This complements existing Insider Threat detection system and improves its performance.

Following solutions to Insider Threat detection problem is contributed during the research:

1. Classification of Insider Threat based upon attack scenario. The Insider Threat is classified based upon the type of attack the insider can perform in the organization. A seven-factor kernel density estimation is developed for the classification.
2. Development of encrypted honey pots for detection of Insider and deployed on a working network. The honey pot comprises of an actual system which forward all system and network calls to the data stream analyzer.
3. Development of real time data stream analyzer to identify the threat posed by the insider. The real data stream analyzer collect the data from the honeypot sensors and identify any Insider Threat based upon system and network calls.
4. Reduced the system configuration requirement as compared to the previous system by introducing honeypot sensor which can be any system or virtual machine in a network.

1.10. Layout of paper

In Section 2 we discuss the existing research work related to insider detection system. Section 3 describes the proposed framework and insider detection mechanism for the factor affecting the organizational security. In Section 4, we discuss the results achieved by proposed classification and finally we conclude the paper with the summary of research work in section 5.

2. Literature Review

This section consists of complete overview of related literature which is very useful for the evaluation of insider detection system. After giving the complete background and structure of insider detection system, we propose an improved and new framework for the assessment of Insider Threat detection. We discussed the literature related to the usefulness of Insider Threat detection, the literature related to different sensor with virtual machine, the literature with reference to sensors and Honeypot along with the activity of behavior based activity recognition and the effect of false alarm on Insider Threat detection system respectively.

2.1. Perspective study on Insider Threat detection system

Mckinney established a by default way of differentiating imposter and insiders in a company by observing the methods of their asset usage [13].This approach was implemented by creating an “ordinary” profile with respect to the run of the mill client conduct. The setting up of these specific profiles was done by the calculations of The Naïve Bayes machine learning. The performance that divert away from the ordinary profiles are considered as extraordinary and possibly suspicious.

So basically, an irregular behavior of profile observed in the machine raise suspicion against that particular profile. This procedure displayed marvelous results in the companies after their implementation with an exact positive ration of 96.7% and the frequency of false positive alarm was found out to be only 0.4%.This information was collected from the company during the duration of three weeks.

Qiao devised a system of Insider Threat detection by the use of battle Insider Threat by actualizing Subject-Verb-Object (SVO) screens [18]. The working of this method is that the procedure is based on the screening out of clients and procedures, catching data, for example, login times, client name, benefit levels, process IDs, and security levels in the framework of the company.

Basically, this method is very effective in pin pointing the possible suspect of Insider Threat in the company. The method enables the server administrator to log into access sorts, for example, peruses, composes, executes, and the "Item" screen recorded document characteristics, for example, proprietor, size, way, and sort. The information that's is collected after these operations is then subjected to thorough investigated from a client metadata archive that contains client information, for example, record framework I/O and procedure movement. The threat which may be present are reviewed with the help of the stride/dread threat model which includes spoofing, tampering, repudiation, records disclosure, denial of carrier, escalation of privilege, damage potential, reproducibility, exploitability, affected users, and discoverability. The results which are achieved after the implementation of this methodology can be related to the results of the instrument that is responsible for the gathering of information about safety measures in an isolated place. However, no results were given that give approximation and direct input about the performance of general framework of the company.

Shavlik devised a hidden recognition framework for the detection of information that has been searched on the framework of the company. The detailed information checks include occasion logs, application information, and client behavioral data, for example, the quantity of running projects and writing rate [19]. The Winnow-based calculation was employed for detection of abnormality discovery it produced and preferred recognition rates over the Naïve Bayes calculation. The working of framework is that it recognizes ay suspicious activity happening on the framework of the company. This method identifies any strange activity in the company's system by running the logged action of one client through the profile of another client. The recognition rates of company were found out to be approximately in the mid to upper 90th percentiles with rates as high as 97.4 percent.

2.2. Sensors with virtual machines

Spitzner employed honeypots and honey tokens as a tool to fight exfiltration insider attacks [20]. Honeypots are basically the PC frameworks in the company which are meant to observe the digital activities of the employees and trap and pinpoint any individual who tries to use the company's data illegally. Honeypots are very helpful in detecting the Insider Threat in the company. Honeypots can instantly detect any irregular activity of employee in the main framework of company and report this irregularity on the spot.

Basically, the movement on a honeypot is thought to be dangerous, and is recorded. Another example of Insider Threat detector are Nectar tokens which are advanced information or data, such a record, a Mastercard number, or a document which are set up to catch any individuals who tries to illegally utilize the data of the company. Till now, no results have been presented regarding the efficiency of this process.

Yu and Chiueh produced the Display-Only File Server (DOFS), it was basically a framework whose purpose was to provide remedy against data theft in 2004 [21]. Working as an independent document server, the DOFS studies that if the client had possessed the authority to manipulate a particular asset of the company. If it is found out that that the client had the right the access the particular asset of the company then the server drops the suspicious level against that client. In this way DOFS hinders people from changing or disturbing the original documents of the company.

Pramanik made use of a security approach that was based on the use of structure like Digital Right. Another methodology for the administration for Insider Threat [17] includes the use of an entrance control structure that requires the client to be checked and secured before the client gets access to peruse/compose/overhaul documents. The drawback is that this framework produces unpleasant effects, for example, it creates problem for clients to continuously open the same document they needed many times to study in a single day.

Park made variation in a Role Based Access Control framework to detect Insider Threat [16] by conducting Composite Role-Based Monitoring (CRBM). Taking into account the benefit of the clients, the following method allows or rejects access to records and assets in view of the client given task. The CRBM make use of threefold structure to characterize access benefits: Application, Operating System, and Organization. The access to assets is allowed or stopped according to the results of these three part structures. However the findings of these experiments did not prove to be really fruitful and did not support cases of an Insider Threat framework that can be used effectively to fight and stop any insider attacks more accurately in companies than the previous already implemented methods and procedures.

Symonenko stated a part based technique in which client parts, setting, and semantics are calculated for the detection of Insider Threat [8]. This method makes use of regular dialect handling to decide themes and territories of enthusiasm inside reports.

These factors are then studied in detail to look at respective client appointed themes and ranges of interest by taking into account the connection of a client participation. Bolster Vector Machines (SVMs) and ontologies are used to direct and observe and client themes and interests. The records were then gathered for the utilization of bunching. The limit of groups and the separation from different bunches are put in use to eliminate the threat.

Cathey made use of the bunching calculations with reference to archives, inquiry questions, and significance appraisals in order to understand any irregularity with reference to company’s interest [5], [12]. In this method, a collection of records are arranged into groups. A client profile is then established with regard to the utilization of bunches. These report groups are then compared and contrasted with the reports in the client’s profile. The significant information which should be known by the organization about the client’s profile is narrated by ordinary client seeking activities and consistent terms inside the records are then forwarded to by the client. The threats are pointed out by finding out the abnormalities in a required report and in the client’s profile as well. The data obtained after the application of this methodology showed that "abuse identification" rates ranges between 90-100 percent and the ratio of false positive alarm fall somewhere around 12 and 15 percent.

Aleman-Meza has carried out research on information rupture which occurs inside the setting of Insider Threat [3]. This research makes use of factual, NLP, and machine learning methods to calculate the significance of got to records regarding insider’s work. Their focus point is in the national security and in the terrorism space. The data of methods and its drafting records are very relevant, steadily dictated, but in terms of linguistic clarification, the data appears to be doubtful, insignificant and indecisive. This brief paper have also shown results that happen to be motivating with a test set of 1000 records.

Liu have shown a method for the detection of Insider Threat in an organization. In this method, the operator breaks down the framework calls and framework call data and traits [11]. They made use of the nearest-neighbor machine learning calculation to find out any abnormal activity in the light of framework calls. The tests proved this method to show a high frequency of false-positive rate for the purpose of Insider Threat detection. If we take into account these results as it is, this conclusion have shown intelligent results for interruption identification.

Anderson suggested a technical engineering based method for the detection of Insider Threat by monitoring the documentation and program occasions [4]. That engineering method consists of sensors and substance based recognition in the given setting. The sensors are basically entities that consist of screens conveyed inside applications (e.g. MS Word and Internet Explorer) and different other working framework. These sensors work on the principle that they thoroughly analyze occasions ad activities, for example, document opening, closing , recoveries, and console inputs. Once the activities have been found out , then these activities are forwarded by a material based- directing framework to the classification of element access control and investigation units. The parts of the entrance control can tally with the project and can detect and prevent the doubtful moves from being made.

2.3. Behavior based activity recognition

Nguyen designed a framework to differentiate suspicious insider activities from regular logins by putting a check on the framework calls which are directly associated with the document framework and procedure [15].This approach studies the calls initiated by the client and also by taking into account the framework calls started by the host framework. The tests have shown that the recorded framework calls begin by the clients reflected a good amount of variation and are not ideal for creating behavioral profiles. The record framework calls were then started again by the working framework, and they showed extraordinary expected performance, and was recommended by the administrators for client behavioral profiles. This research did not show the results about the changes that occurred in the client behavioral profiles. This is beneficial for this research because it gives idea of this concept with respect to the deal that if the desired performance is dependent upon client action or whether the working framework acts randomly, in case of clients as well. The fact is that the framework calls were supposed to generate marvelous results for showing client's performance. After the regular following of all the procedures, 92% of clients have shown a variation of rundown of documents and a reasonable number of them got to records in examples, and 92% of them had a settled rundown of formulated youngster forms. This procedure stressfully reflected all kind of flood attacks along with those in which the methods did not have an altered number of baby procedures.

Another behavior based insider detection model is presented by researchers [2]. The advantages of behavior-based approaches are that it very accurately and remotely finds surprising vulnerabilities which cannot be found without it. They even contribute substantially to the (partially) automatic discovery of those new attacks on the systems. They are less passionate about operative system-specific mechanisms. They conjointly facilitate discover 'abuse of privileges' sorts of attempts that don't really involve exploiting any security vulnerability. In short, this is often the paranoid approach: Everything that has not been seen historically in the past is dangerous.

Behavior-based intrusion detection techniques assume that associate intrusion are often detected by observant a deviation from traditional or expected behavior of the tactic or the users. The model of traditional or valid behavior is extracted from reference data collected by varied means that. The intrusion detection methodology later compares this model with this activity. Once a deviation is ascertained, associate alarm is generated. In different words, something that doesn't correspond to a historically in the past learned behavior is believed regarding intrusive. Therefore, the intrusion detection methodology may be complete (i.e. all assaults got to be caught), however its accuracy may be a troublesome issue (i.e. you get many false alarms).

The warning rate being very high is typically cited because the main disadvantage of behavior-based techniques as a result of the complete scope of the behavior of associate data methodology might not be coated within the work of the educational part. Also, behavior alters over with time, introducing the need for periodic on-line preparation of the behavior profile, ensuring either in inaccessibility of the intrusion detection methodology or in further false alarms. The data methodology endure assaults at the same time the intrusion detection methodology is learning the behavior. As a result, the behavior profile contains intrusive behavior, that isn't detected as abnormal. The behavior Driven hybrid model is represented in Fig. 1.

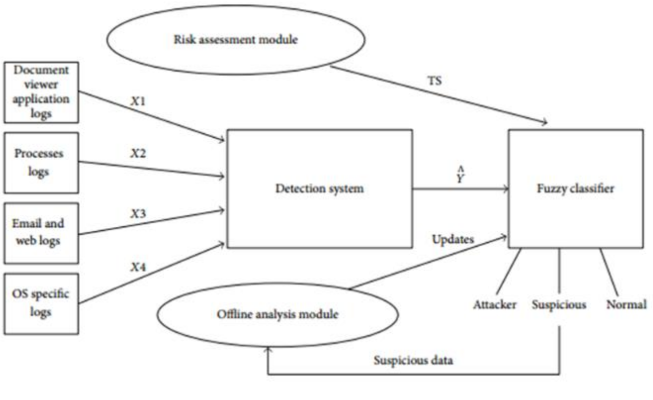


Fig. 1- Insider Threat Detection and Prevention Framework [1]

2.5. Effect of false alarm on Insider Threat detection system

Hybrid approaches uses two or more different techniques for detecting insiders, the models derived from hybrid approaches possess the characteristics of the derived models. An example of hybrid information security model can be seen in "Using Genetic Algorithm to Minimize False Alarms in Insider Threats Detection of Information Misuse in Windows Environment," developed by researchers [1]. Which uses evolutionary algorithms with behavior driven models to reduce amount of false positive alarm raised by behavior driven information security models?

Hybrid approaches have many advantages over the previous models but one issue remains a problem. Every organization have a different thresh hold value for detecting the insider to compensate that the insider detection system requires a lot of configuration in a lot of different scenarios to get fruitful results. Without suitable configuration the performance of hybrid insider detection system cannot be satisfactory. This creates the need of system which require little or no configuration in different scenarios.

3. Materials and Methods

3.1. Background

The method used in this research work is stated below. The population of the present research is all the computer systems of TEST BED lab set-up. The sample consists of 50 PC systems of TEST BED which is a complex network of PCs servers in a computer lab of random organization. A huge number of employees, interns, and researchers at TEST BED agreed to be checked for a pre-determined period of time on experimental basis. Randomness was a must component for all participants. However, their part within TEST BED (employee, researcher, intern) was allocated specifically. All experiments were performed on the Windows 7, 8.1 and 10 operating system.

All participants have adequate knowledge that their file system, network, hardware, and process information would be checked and that data would be studied on the basis of the user and system activity on their respective computers. Although the participants got the idea that their digital activities are being checked, we enquired from them that if they would carry on their usual tasks on PC normally the way that they used to conduct if their behavior was not being checked. Issues and problems regarding the accuracy of this supposition are discussed in Chapter 4 along with other ethical and legal issues regarding the present work. The data was gathered form the participants over periods of time that consisted of a duration of few days to a few weeks depending upon the test.

3.2. Honeypot Sensor Setup

The honey pot comprises of an actual running window or linux which operating system which can be running on a physical machine or virtual machine. The honeypot is configured to forward logs to the data stream analyzer. In the Test Bed, a honey pot sensor is placed in the active directory of the network in which all systems are connected in the network used active directory for their system communication over the network so every activity on the system can be logged easily .The malicious system call are forwarded to a read data stream analyzer which is running on a separate system in which the data streams from various sensors is represented graphically based upon the threat level. The data streams contain system logs which helps to clearly identify which malicious activity is going on the system.

3.3 Data Acquisition

For Windows based environment, data collection was conducted by use of a monitoring tool based upon Microsofts.NET framework which is called IBM WinCollect. The function of this tool is to monitor and gather data depending upon the events and states of the operating system and also depends upon the changes in the hardware respectively. The data sets and variables are associated with Install Wincollect agent and configuration console in windows and configure it as shown in screenshot. To send logs on UDP, they create destination in UDP Section and vise versa.

First one has to select which type of events one would like to send like “Local System”, “Security”, ”Application” etc.

Here Device Address is the machine IP address. One can Add destination, which is just formed, at the bottom of this window in destination box by clicking on “Add”.

For Linux based machines, rsyslog daemon was used for the acquisitions of logs, the configuration takes place in the following manner

```
$ sudo nano /etc/rsyslog.conf
```

Add the following configuration at bottom

```
$template linuxbox,"<%pri%>%timestamp% 192.168.2.50 %syslogtag%%msg%"          ## IP of machine
# Use only one UDP or TCP depends on the importance of device
*.*     @192.168.2.50:514;linuxbox       ## For UDP
*.*     @@192.168.2.50:514;linuxbox   ## For TCP
$ sudo service rsyslog restart
```

3.3.1 Profile Training Phase

The experiment starts with the profile training phase. This profile training phase consists of gathering and processing of all the activities conducted by the participant, processes in hardware, processes in network and in the file system data to create a normal profile. The collected data is then processed with the help of k-means clustering and kernel density estimation algorithms. Each normal profile is represented by the seven kde distributions beneath:

Processor usage profile – kde possibility distribution relies upon the tactics of person data in the stage of profile education.

Memory utilization profile – kde possibility distribution relies upon the approaches of user information inside the level of memory schooling.

Hard drive utilization profile – kde possibility distribution is dependent upon the techniques of person difficult drive data inside the Section of profile training.

Process threads profile – kde probability distribution is depending on the frequency of energetic threads of each process during the level of profile education.

File machine profile – kde possibility distribution relies upon the data acquired from the activity of report device hobby accumulated inside the degree of profile schooling**.**

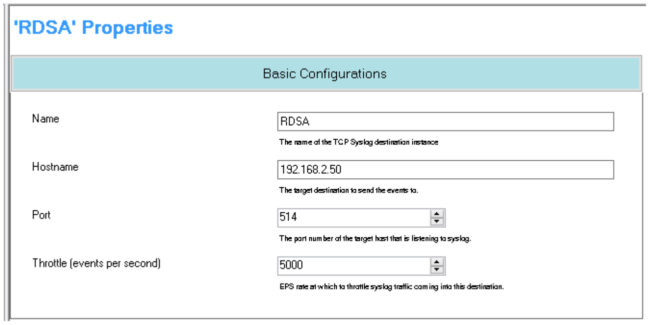
Network IP profile – kde opportunity distribution is dependent upon ip addresses and community connection statistics that are acquired in the level of profile education.

Community port profile – kde probability distribution is dependent on the records of network port usage which is obtained in the degree of community schooling

3.3.2 Nominal Behavior Probability

A KDE can be described as a collection of probability distribution. In the view of preset research work, when the probability distributions are associated with the data of normal profile, then the end result is that the user behavior is normal. Thus the user behavior can be termed as nominal behavior probability

In Fig. 2, we illustrated the participant process profile. It can be seen from the Figure that in the stage of profile training, the frequency of use of CPU normally fell between 15-40 percent. Following the creation of the frequency of use of processor falls between 15-40 percent. The value of this percentage shows comparatively huge probability of nominal behavior. The values of the use of processor outside the range of 15-40 percent result in comparatively less probabilities of nominal behavior.



'RDSA' Properties

Basic Configurations

Name: RDSA
The name of the TCP Syslog destination instance

Hostname: 192.168.2.50
The target destination to send the events to.

Port: 514
The port number of the target host that is listening to syslog.

Throttle (events per second): 5000
EPS rate at which to throttle syslog traffic coming into this destination.

Fig. 2 - Honey Pot Sensor Configuration Using IBM Win get

3.4. Real Data Stream Analyzer

The real data stream analyzer is an Ubuntu based system which is placed on a separate machine. The real data stream analyzer performs the following functionalities.

3.4.1 Sensor Data Parser

This module parse incoming data streams from honeypot sensors in a readable format. The data generated from the honey pot sensor are in form of big Java Sting Object Notation strings which is very difficult to read. The module separates the strings in form of source IP, destination IP, event name, user name, Date, Time and many other important are factors represented in a readable form. After collecting all the relevant data of the event, the data is forwarded to the threat analyzer to identify potential Insider Threat. The threat analyzer is a separate module and is placed on the same server.

3.4.2 Threat Analyzer

The threat analyzer analyze the parse data on three different levels. Each level is designed to filter out any chance of a false alarm. It only sends alert notification when it is unable to detect event activity an any level details of which is explained in the algorithm below:

Algorithm 1: False Positive Insider Threat Detection

```
Step1: If user event activity is normal then no alarm
Step2: Else check system running software and application events
Step3: If user activity is according to software and application events then No Alarm
Step4: Else check system hardware changes events
Step5: if hardware changes are allowed to user then No Alarm
Step6: Else check user privileges
Step7: if user modifies files as per user privileges then No Alarm
Step8: Else raise the alarm.
```

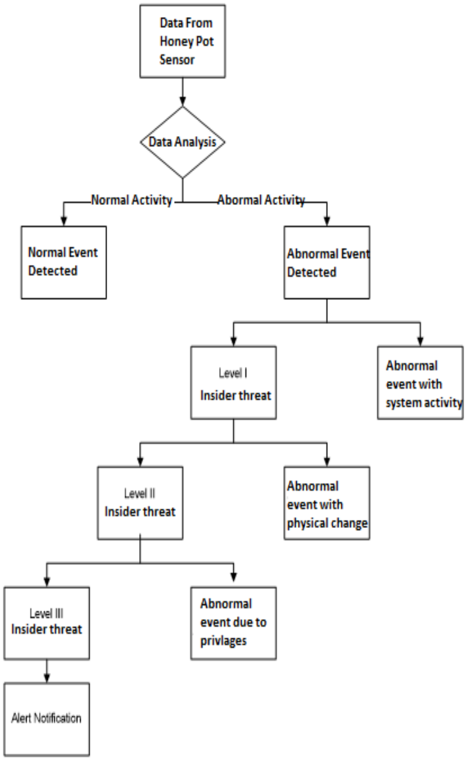
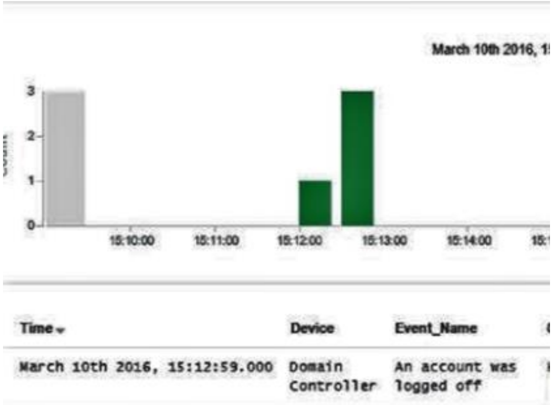


Fig. 3 -Three level classification of threat analyzer algorithm

The algorithm is simple enough to be easily implementable and is effective enough to detect any false alarm generated by user system activity. The three level classification of user system activity first detect that if the event is software or system service generated which is previously identified in profile training phase then it ignores the event. Next if any hardware changes occur in the system which generated suspicions events should be analyzed and if the hardware changes events fall in normal user behavior generated events then it ignores the events. Finally it checks the changes in data made by the users. If the data modified by the user is according to the given user privileges then it ignore those events. Any suspicious events that are generated by abnormal thread, read, writes, network logins, data transfer, hardware changes, and privileges changes are detected in real time for the identification of Insider Threat. The three-level hierarchal classification of proposed threat analyzer algorithm is shown in Fig. 3. The detected Insider Threat is displayed in a user-friendly web based GUI which is only accessible to system administrators details of which is given in the next Section.

3.4.3 Threat Representation in GUI

A web-based GUI is developed for the representation of the insider. The GUI is built with Kibana frame work which is used in the representation of big data. As the sensor generated logs ranges in GB's per hour so accurate representation of important event logs was necessary. The data is represented in two formats first in the form of an event graph which is update with time and then the logs represented in a customizable manner so system admin can choose what type of information they wanted to be displayed first. All logs are stored in a database which are retrievable by log search functionality and can be used for investigation if an incident happens. Fig. 4 and 5 show the actual GUI of web platform.



Time	Device	Event_Name
March 10th 2016, 15:12:59.000	Domain Controller	An account was logged off

Fig. 4 - Representation of System Logs in Web based GUI

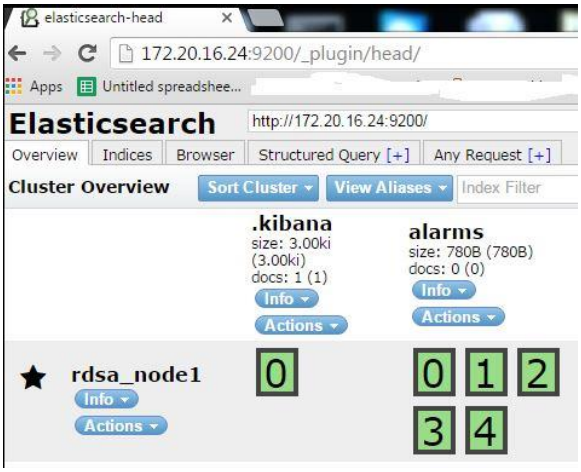


Fig. 5 -Representation of System Logs in Web based GUI

3.5 Test Cases

In the given test cases, the conduct of no participant was observed to be threatening. This result was expected by the researcher as the sample participants involved in the study were fully aware of the fact that that their behavior is being continuously monitored. By considering this factor, the researcher also carried out tests by prompting ordinary insider assaults with the objective of detecting any irregular conduct during the simulated attack. The descriptive Section below shows the different experimental evidence that were collected at the end of each research. The researcher would also discuss the objectives and issues related to every experiment in this Section.

3.5.1 Typical Behavior

This test made use of contrast of normal behavior of users to their profile of normal behavior. At the beginning of the experiment, the participants were directed to create a normal behavior profile. Afterwards, their activities were observed in the view of probabilities of nominal behavior while they were performing their normal day to day jobs. However, this practice does not necessarily relate to an insider attack but in a way, it somehow give worthy understanding of the desired probabilities for nominal behavior which is supposed to be normal.

3.5.2 Thread Bomb

A thread bomb is largely a type of assault of denial of carrier. On this experimental check, a thread bomb is added into an application together with notepad or net explorer and it starts off evolved to generate both boundless quantity of threads and a pre-calculated variety of threads. This process appears after the level of profile education. Like diverse denial of offerings attack, this particular attack is also much unexpected and has the ability to shut down all the functions of the host machinery making it cripple, stopping its functions and making the host vulnerable to various forms of assaults. It is therefore very important to detect this attack and take precautionary measures earlier before it damages the system of the host and make it disable .In order to conduct this test, the researcher initiated an attack that exploited Notepad and created two thousand threads.

The abnormal activity is quickly detected by the Insider Threat detection system and displayed on the real data stream analyzer.

3.5.3 Abnormal File System Activity

Instances of cybercrime exfiltration and data corruption consists of admittance to the working of the file system. In the cases of real life exfiltration, the hacker sometimes access directories that they are not allowed to access and which comes out of range of their work ethics. The researcher conducted the following experiments for the detection of abnormal activities in the file system of TEST BED.

Abnormal File Deletions: Prior to the creation of profiles of normal users, the researcher formulated random 40 MB directories within the programs of users, and formulated 32 directory in the system. After the formulation of normal profile, it was observed that many participants methodologically deleted a group of files from these respective directories. The participants narrated that they rarely execute tasks within the boundary of system 32 and in program directory as they make use of machines with least user privileges. (LUP).

Abnormal File System Activity: After formulating a normal profile, each participant formulated a varied version of changed, deleted, and modified files in the drives that they usually do not access.

3.5.4 Abnormal Network Activity

In the given experiments, the researcher has attempted to detect irregular endpoint-to-endpoint community interest through investigating the IP addresses of network connections. The researcher has also attempted to look at the ordinary port get right of entry to. The subsequent experiments have been performed after the degree of profile training:

New Network Connection: Every participant can pay everyday visits to an overseas internet site that they generally do now not go to.

New Port: Each participant tried to connect up with the remote servers on ports that they normally do now not connect.

To simulate the issue the researcher, launch a simulated brute force attack for remote logging in on a machine present in the network.

3.5.5 Abnormal Process Activity

The activity of abnormal process can be a precursor that dangerous application, malicious code, or any other form of malware is there in the system. As, it has been narrated earlier that the insider attacks on the organization can prove to be very harmful and dangerous if they are left undetected and uninvestigated. The researcher has invented different experiments that could be really beneficial in detecting the abnormal process activity. The following tests were executed after the stage of profile training:

New Program: Each participant began to utilize the program that they did not apply in the stage of profile training

3.5.6 Abnormal Hardware States

Atypical hardware states can be a sign of a range of denial-of-service and records corruption assaults. The following assessments were depended on to be helpful in finding strange strategies and hardware changes:

Processor: Each participant conducted and utilized programs to improve the work load of the processor.

Hard Drive Test 1: After the completion of stage of profile training, every participant downloaded at least 10 GB of data into the disk drive.

Hard Drive Test 2: Earlier than the final testing of level of profile training, each participant downloaded 15 GB of random information into the disk drive of employee system. After training segment, the 15 GB of information downloaded with the aid of the participant on their systems would be deleted.

4. Results and Discussion

The proposed framework which has been explained in the section 3, shows feature extraction of different daily threat activity with insider data from embedded sensors of networks. The honeypot sensor detects the abnormal activity of an individual and the sensor show the threat posed by the individual. The Weka machine learning toolbox is used for feature extraction of raw data from system calls generated from honeypots. The framework is used for 3-level hierarchical classification.

In this section we have evaluated the performance of experiment and subsequent result for Insider Threat detection with activity classification and hierarchical detection of insider, an experimental study of 50 users have been carried out with mentioned dataset in this research as shown in Table 1. In the present experiment, raw data is collected with the help of network sensors. The raw data of honeypots is transformed to segment of 10 seconds called sample data. Then raw data is used for further feature extraction of different classes in daily life activity. The feature classification is used with Insider Threat value for further 3 classes i.e. Class1, Class2 and Class3. The class1 contains normal user behavior in the network like with Insider Threat value ranging from 0 to 1. The Class2 contains suspicions user behavior like accessing unauthorized data with Insider Threat value ranging from 2 to 5. Similarly the Class3 contains insider attacker behavior like manipulating unauthorized data with Insider Threat value ranging from 6 to 10.

4.1 Validation of Honeypot Sensor

The validation of honeypot sensor is done by generating a single Insider Threat in the TEST BED of 50 systems and then confirming its detection by the insider detection system. The standard deviation and mean of acquisition of Insider Threat values from normal user behavior generated and manually generated attacks from sensor detected data is calculated

4.2 False Positive Detection At Level 1

The detection of false positive alarm is based on user activity factor. Insider Threat detection percentage are checked in accordance with system activity i.e. from normal to suspicious; the abnormal Insider Threat is detected as false positive alarm due to different

Application software and system services running on system which raised false alarms as shown in Table 1.

4.3 False Positive Detection At Level 2

The abnormal insider is effected by the physical system changes like USB connection, peripheral device removal etc. The level 2 classification is used to detect false positive Insider Threat detection percentage due to physical changes in system as shown in Table 2.

4.4 False Positive Detection At Level 3

The increase of Insider Threat detection percentage is due to the activities of a privileged user who has the write of read, write and modify the system files this caused unnecessary false Insider Threat alarm which can be seen in Table 3.

4.5 Actual Abnormal Insider Threat Value

Table 4.1 is used to detect true positive result based on Insider Threat detection percentage. By classification of a 3-level system, true positive Insider Threat value is detected as shown in Table 4.

User	Scenario	User Activity	Initial Category	Final Category	False Alarm %
6	Abnormal Process Activity	System administration service was running	Attacker	Attacker	100%

12	Thread Bomb	Data compression software was	Attacker	Attacker	100%
18	Abnormal Process Activity	System administration service was running	Attacker	Attacker	100%
24	Abnormal Network Activity	Torrent Client was running on system	Attacker	Attacker	100%
25	Abnormal Process Activity	System administration service was running	Suspicious	Attacker	100%
48	Thread Bomb	Data compression software was	Attacker	Attacker	100%

Table 1 Insider Threat Alarms Due to Application Software and System Services

User	Scenario	User Activity	Initial Category	Final Category	False Alarm %
9	Abnormal Hardware State	USB Disk Attached on System	Suspicious	Attacker	100%
14	Abnormal Hardware State	Data Moved from one drive to another within the system	Attacker	Attacker	100%
16	Abnormal Hardware State	USB Disk Attached on System	Suspicious	None	100%
34	Abnormal Hardware State	Data Moved from one drive to another within the system	Attacker	Attacker	100%

Table 2 Insider Threat False Alarm due to hardware Changes

User	Scenario	User Activity	Initial Category	Final Category	False Alarm %
49	Abnormal File System Activity	File is deleted by a privileged user	Suspicious	Attacker	100%

Table 3 Insider Threat False Alarm Due to Activities of Privileged Users

4.6 Result Comparison with Existing System

Table 4 shows the difference in results between proposed and existing system for one-hour operation which is graphically represented in Fig. 7. Table 6 shows the difference in results for 24 hours of operation which is graphically represented in Fig. 8. It is clearly seen that the proposed system is working more reliably as compared to the existing system. Table 7 shows the difference of results in proposed and existing systems for the period of 10 days which is graphically represented in Fig. 9. From the gather data of 10 days total number of malicious events was found to be 4963 for existing system and 3874 in proposed system. It is then calculated average improvement in proposed system is 21.9% as compared to existing system

This research made use of the procedure of Insider Threat detection with the use of honeypot sensors. In local networks, honeypots sensors are used to detect and relay the information of Insider Threat with the activity recognition of an insider. The Real Data Stream analysis plays the role of hub, processor and sensor, to transmit the preprocessed data received from honeypot sensor. The classification of data from honeypot sensor is processed by using the Insider Threat rate classifier in Kabana toolbox for activity recognition.

User	Scenario	User Activity	Initial Category	Final Category	False Alarm %
5	Thread Bomb	Simulated Thread Bomb Launched on system	Attacker	Attacker	0%

8	Abnormal Network Activity	Simulated Port Scanning was performed from the system	Attacker	Attacker	0%
19	Thread Bomb	Simulated Thread Bomb Launched on system	Attacker	Attacker	0%
28	Abnormal Process Activity	Simulated Malware was introduced in to the system	Attacker	Attacker	0%
29	Abnormal Process Activity	Simulated Malware was introduced in to the system	Attacker	Attacker	0%
30	Thread Bomb	Simulated Thread Bomb Launched on system	Attacker	Attacker	0%
33	Abnormal Network Activity	Simulated Port Scanning was performed from the system	Suspicious	Attacker	0%
37	Abnormal File System Activity	File is Copied by a under privileged user	Attacker	Attacker	0%
38	Abnormal Process Activity	Simulated Malware was introduced in to the system	Attacker	Attacker	0%
42	Thread Bomb	Simulated Thread Bomb Launched on system	Attacker	Attacker	0%
44	Abnormal Hardware State	Data Copied to external hard drive from the system	Suspicious	Attacker	0%
46	Abnormal Process Activity	Simulated Malware was introduced in to the system	Attacker	Attacker	0%

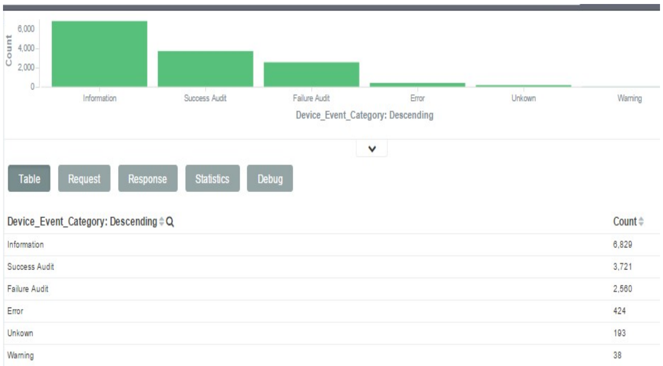
Table 4 Real Insider Threat Detected

The real time monitoring of system calls by using honeypot is low cost and less complex, because the system calls of measurement from each system is not required to be implanted and monitored continuously. Results have shown that the present framework, activity classification and false positive Insider Threat value detection mechanism is producing accurate results. The zero mean error is used to count Insider Threat value with the real time accuracy.

In near future, we aim to extend his work with data efficiency and other relative vital sign for measurement and analysis of Insider Threat.

Fig. 6 shows the actual number of warnings generated by the proposed Insider Threat detection system. In the graph it can be seen that out of nearly eight thousand events only thirty-eight are found to be actual warning. This is achieved by three level hierarchal classification of user generated events to avoid false alarm and optimize system performance.

Fig. 7 shows the data of Table 5 in a graphical manner, in which effects of different events generated by users are measured against both proposed and existing systems. Events generated by thread bombs, abnormal network activity, file read and write operations, hardware changes and system process activity are carefully measured to calculate the performance improvement in proposed system as compared to existing system.



Device_Event_Category: Descending	Count
Information	6,829
Success Audit	3,721
Failure Audit	2,560
Error	424
Unkown	193
Warning	38

Fig. 6 - Actual number of warnings generated by Insider Threat detection system

Events Activity	Proposed System	Existing System
Thread Bomb	20	24
Network Activity	19	48
File Read and Write	28	35
Hardware Changes	27	28
Process Activity	20	25

Table 5-Event Activity Comparison between Proposed and Existing System in an Hour.

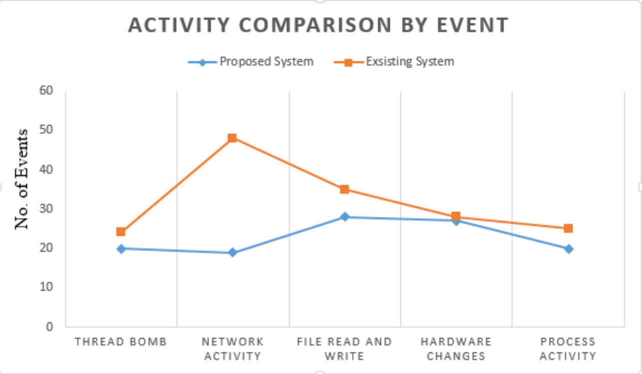


Fig. 7 -Activity Comparison of Proposed and Existing System with Respect to Events

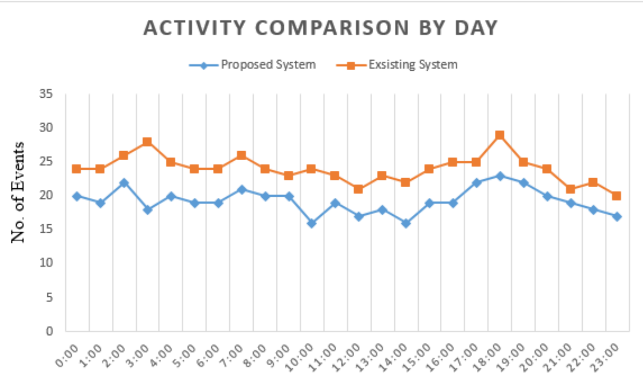


Fig. 8 -Activity Comparison of Proposed and Existing System with Respect to Time

Fig. 8 shows the data of Table 6 in a graphical manner, in which performance of existing and proposed systems are compared for a period of one day with respect of events generated to compare the insider detection performance of both systems. 16 % improvement was measured in a single day of comparison between both systems

Fig. 9 shows the data of Table 7 in a graphical manner, in which average performance of Insider Threat detection system of existing and proposed system were compared for the period of ten days. It was calculated that average performance increase in proposed system is 21.9 % as compared to existing system.

Time	Proposed System	Existing System
0:00	20	24
1:00	19	24
2:00	22	26
3:00	18	28
4:00	20	25
5:00	19	24
6:00	19	24
7:00	21	26
8:00	20	24
9:00	20	23
10:00	16	24
11:00	19	23
12:00	17	21
13:00	18	23
14:00	16	22
15:00	19	24
16:00	19	25
17:00	22	25
18:00	23	29
19:00	22	25
20:00	20	24
21:00	19	21
22:00	18	22
23:00	17	20

Table 6 Event Activity Comparison between Proposed and Existing System in a Day

Date	Proposed System	Existing System
15-Jul-16	463	576
16-Jul-16	220	440
17-Jul-16	216	380
18-Jul-16	398	496
19-Jul-16	458	587
20-Jul-16	446	523
21-Jul-16	396	476
22-Jul-16	410	482
23-Jul-16	416	483
24-Jul-16	424	520

Table 7 Event Activity Comparison between Proposed and Existing System in 10 Day

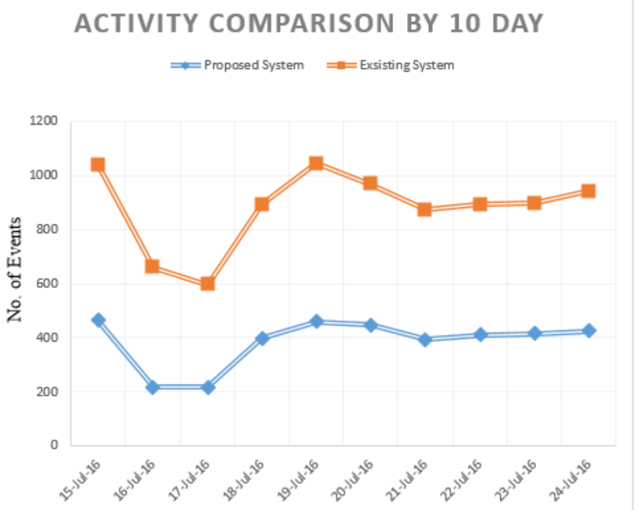


Fig. 9 -Activity Comparison of Proposed and Existing System with Respect to 10 day.

5. Conclusion

Section 1 states the background and significance of Insider Threat detection. Insider Threats are very common in big multinational companies and their rival companies can easily destroy other company with the help of an insider. An insider is basically a part of the parent company and works for the benefit of rival company. An insider can damage the data of company with the help of exfiltration, data corruption, and abnormal activities and by denial of services technique. Therefore, it is very important that companies should be equipped with adequate Insider Threat detection system to ensure safety from the Insider Threat. Insider Threat can be check by the use of sensors and honeypots who can check the activities of employees on the main framework and raise an alarm if any abnormality is found out in the behavior of employees.

Section 2 is about the literature review of term Insider and Insider Threat detection. We defined the term Insider in detail and quoted many examples and cases of Insider assaults in various organizations. The bottom line of this chapter is that Insider Threat is a very recent and contagious type of cybercrime in the modern era and therefore, proper safety measures should be adopted to cope with the bad and harmful effects of this problem.

In Section 3, we proposed the framework and design for the research work. The researcher has taken sample of 50 PCs in a TEST BED lab system. The research has continuously monitored the activities of employees within a specific duration. The nominal behavior of sample population was studied before and after the profile training phase. Basically, we implanted a honeypot on network and that honeypot collected data of any person who try to illegal access any irrelevant website. The honeypots have the ability to track both the IP address and the physical address of the attacker.

In Section 4, we presented the validation of Honeypot sensors with reference to the detection of Insider Threats. The Insider Threats have been validated and classified on three levels respectively. The research has graphically represented the validation of Insider Threat detection with reference to honeypots in the form of Tables and illustrations. These Tables and their values have been extracted after the study of sample population of TEST BED participants; activities on PC frameworks. The improvement in proposed Insider Threat detection system is carefully calculated and average improvement of 21.9 % was found as compared to existing system.

Finally, we presented the findings of the results after the quantification of collected data. We gave a brief review of insider and Insider Threats to the readers. Consequently, we pointed out various cases of insider attacks in different organizations and concluded that these insider attacks could have been prevented if the companies have implemented in them a proper insider detection system. Then, we mentioned some old techniques of detecting Insider Threats in organizations that made use of Honeypots. In previous times, Honeypot sensors were used to be installed at every single PC and most of the honeypots detect false positive alarms about the insiders. This created a lot of complexities. Therefore, the researcher took motivation from this point and tried to improve this process of Insider Threat detection by associating the honeypot on the main network of PC server. We then made software and the purpose of this software was that the honeypot keep a check on every activity of each

PC on the system. Whenever any suspicious activity was found out by the honeypot, then the system of that suspicious employees locked automatically. This technique is produced for the detection of Insider Threat. It is expected that in future more research would be done to improve the system of Insider Threat detection system by optimizing the amount of data generated from each honey pot to save disk space and to avoid unnecessary network activity. Kernel Density Estimation algorithms can also be improved to make the insider detection system appropriate for external threats detection and to perform threat intelligence

REFERENCES

1. Ahmad, M.B,, Akram, A., Asif, M. and Ur-Rehman, S., 2014. Using genetic algorithm to minimize false alarms in Insider Threats detection of information misuse in windows environment. Mathematical Problems in Engineering, 2014.
2. Ahmad, M.B., Akram, A. and Islam, H., 2013. Implementation of a behavior driven methodology for Insider Threats detection of misuse of information in windows environment. International Information Institute (Tokyo). Information, 16(11), p.8121.
3. Aleman-Meza, B., Burns, P., Eavenson, M., Palaniswami, D. and Sheth, A., 2005, May. An ontological approach to the document access problem of Insider Threat. In International Conference on Intelligence and Security Informatics (pp. 486-491). Springer Berlin Heidelberg.
4. Anderson, K., Carzaniga, A., Heimbigner, D. and Wolf, A., 2004. Event-based document sensing for Insider Threats. University of Colorado, Computer Science Technical Report CUCS-968-04.
5. Cathey, R., Ma, L., Goharian, N. and Grossman, D., 2003, November. Misuse detection for information retrieval systems. In Proceedings of the twelfth international conference on Information and knowledge management(pp. 183-190). ACM.
6. Grobauer, B. and Schreck, T., 2010, October. Towards incident handling in the cloud: challenges and approaches. In Proceedings of the 2010 ACM workshop on Cloud computing security workshop (pp. 77-86). ACM.
7. Mallah, G.A. and Shaikh, Z.A., 2005. A platform independent approach for mobile agents to monitor Network vulnerabilities. WSEAS Transactions on Computers, 4(11), pp.1672-1677.
8. Ali, G., Shaikh, N.A. and Shaikh, Z.A., 2008, April. Towards an automated multiagent system to monitor user activities against Insider Threat. In Biometrics and Security Technologies, 2008. ISBAST 2008. International Symposium on (pp. 1-5). IEEE.
9. Hayden, M., 1999. The Insider Threat to US government information systems (No. NSTISSAM-INFOSEC/1-99). NATIONAL SECURITY AGENCY/CENTRAL SECURITY SERVICE FORT GEORGE G MEADE MD.
10. Legg, P.A., Buckley, O., Goldsmith, M. and Creese, S. (2015) ‘Automated Insider Threat detection system using user and role-based profile assessment’, IEEE Systems Journal, ,pp. 1–10.
11. Liu, A., Martin, C., Hetherington, T. and Matzner, S., 2005, June. A comparison of system call feature representations for Insider Threat detection. In Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop (pp. 340-347). IEEE.
12. Ma, L. and Goharian, N., 2005, March. Query length impact on misuse detection in information retrieval systems. In Proceedings of the 2005 ACM symposium on Applied computing (pp. 1070-1075). ACM.
13. McKinney, S. and Reeves, D.S., 2009, April. User identification via process profiling. In Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies (p. 51). ACM.
14. Moore, Andrew P., Dawn M. Cappelli, Thomas Caron, Eric Shaw, and Randall F. Trzeciak. "Insider theft of intellectual property for business advantage: a preliminary model." In Proc. Of the 1st International Workshop on Managing Insider Security Threats (MIST2009), Purdue University, West Lafayette, USA. 2009.
15. Nguyen, N.T., Reiher, P.L. and Kuenning, G.H., 2003, June. Detecting Insider Threats by Monitoring System Call Activity. In IAW (pp. 45-52).
16. Park, J.S. and Ho, S.M., 2004, June. Composite role-based monitoring (CRBM) for countering Insider Threats. In International Conference on Intelligence and Security Informatics (pp. 201-213). Springer Berlin Heidelberg.
17. Pramanik, S., Sankaranarayanan, V. and Upadhyaya, S., 2004, December. Security policies to mitigate Insider Threat in the document control domain. In Computer Security Applications Conference, 2004. 20th Annual (pp. 304-313). IEEE.
18. Qiao, H., Peng, J., Feng, C. and Rozenblit, J.W., 2007, March. Behavior analysis-based learning framework for host level intrusion detection. In 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'07) (pp. 441-447). IEEE.
19. Shavlik, J., Shavlik, M. and Fahland, M., 2001, October. Evaluating software sensors for actively profiling Windows 2000 computer users. In Fourth International Symposium on Recent Advances in Intrusion Detection.
20. Spitzner, L., 2003, December. Honeypots: Catching the Insider Threat. In Computer Security Applications Conference, 2003. Proceedings. 19th Annual(pp. 170-179). IEEE.
21. Yu, Y. and Chiueh, T.C., 2004, October. Display-only file server: a solution against information theft due to insider attack. In Proceedings of the 4th ACM workshop on Digital rights management (pp. 31-39). ACM.