AN ANALYSIS OF CURRENT COMPUTER NETWORK ATTACK PROCEDURES, THEIR MITIGATION MEASURES AND THE DEVELOPMENT OF AN IMPROVED DENIAL OF SERVICE (DoS) ATTACK MODEL

IHEKWEABA OGECHI, INYIAMA H.C, IHEKWEABA CHUKWUGOZIEM

***ABSTRACT* - This paper presents the major network attack profiles, taxonomies, classification and identification frameworks.**

Further, current approaches for intrusion detection on networks were x-rayed though, with emphasis on Denial of Service (DoS) attacks only.

The concept of intelligent agents and subsequently adaptive systems was elucidated, the properties derived were used as a proof of concept, and then applied in the development of a model for an adaptive intelligent multi-agent for network protection, against denial of service attacks.

***KEYWORDS:* Agent, Denial of Service attack, Dynamic Host Configuration Protocol, Transmission Control Protocol, / Internet Protocol**

1.0 INTRODUCTION

As the world has moved further into information – driven global economy, the value of information, and controlled access to that information, has never been greater [1]. The goal of IT infrastructure therefore is to create systems that can detect and protect against unauthorized access while providing timely access to legitimate users. Today’s network must be able to respond to attacks in ways that maintain network availability and reliability and allow a business to continue to function. In many respects, the goal of security is to make networks more resilient by making them more flexible. Rather than succumb, networks must be able to absorb attacks and remain operational, much in the same way the human immune system allows us to keep functioning in the presence of viruses and related bacterial infections.

The future of security technology has changed more in the last three years than it did in the prior ten years [2]. The extent of these changes, as well as the rate of changes, has made it difficult for security IT departments to keep up with these challenges. Before IT departments can regain control, they must better understand the changing technological landscape.

A network can no longer be secured by simply securing the network perimeter [2], since corporations have consolidated their data centers, converged internal networks, and embraced the internet.

Technically, the kinds and nature of security planning adopted have to correlate especially with the attack profiles such as:

Application layer attacks, auto rotes, backdoors, man – in –the – middle attacks, network reconnaissance, packet sniffers, password attacks, brute force attacks, port redirection attacks, trojan horse attacks and viruses, trust exploitation attacks, Denial of Service (DoS) and Distributed Denial of service (DDoS) attacks [3].

Denial of Service attacks is a major cause of incorrect operations in the internet and is arguably one of the most serious threats that the internet community faces today [4], [5].

From the time it is detected and recovered from, the victim is virtually paralyzed and cannot respond to legitimate requests. For large commercial sites, this translates to losses of billions of dollars in magnitude [6].

Corporate networks, and the attacks used to exploit them, are so complex that no single mechanism can be relied upon to keep them secure. This has led to the concept of “Defense

in Depth” [2]. Until recently, this concept had been built on the notion of proactive defense. The network security community, in the spirit of postulations of this work, has begun looking to seemingly unrelated real – world examples such as human` immune system as a model for the self – defending network. Other real – world systems that have also proven to be instructive can be found in the field of epidemiology and in the way communities police themselves. A common theme with all of these systems is that they employ adaptive as well as proactive defenses.

The self – defending network based on adaptive intelligent agent monitoring, provides systems – based solutions that allow organizations to use their IT infrastructure in new ways to reduce windows of vulnerability, minimize the impact of attacks, and improve overall infrastructure availability and reliability [2]. This helps create autonomous systems that can quickly react to an outbreak with little or no human interaction. This type of rapid response is required to thwart the latest forms of misuse that are much more virulent than their predecessors.

2.0 BASIC NETWORK ATTACK CHRACTERISTICS

The section reviews discussion on DoS under two broad categories: flood attacks and malformed packets. Understanding the taxonomy of DoS attack is essential in thinking about appropriate counter measures. In his book [3] Tood Lammle describes a number of attacks, some of which are mentioned below:

Autorooters: Crackers use something called a rootkit to probe, scan and then capture data on a strategically positioned computer that is poised to give them “eyes” into an entire system [3].

Back doors: These are simply paths leading into a computer network. Though simple invasion or via more elaborate “Trojan horse” code hackers can use their implemented inroad into a specific host or even a network whenever they want to [3].

Other attack profiles he described [3] are man – in –the – middle attacks, network reconnicaissance, packet sniffer, password attacks, brute force attacks, port redirection attacks, Trojan horse attacks, viruses and denial of service (DoS) attacks.

A Denial of Service attack on a network could take one of the two possible forms. A malicious party (a.k.a the attacker) would cause the network not to transmit data which it naturally should be sending to any of its bona fide clients. On the other end of the spectrum, the network could be caused to send spurious messages, which are not needed by the eventual recipient. By far, the most common form of DoS in today’s networks involves creating excessive bogus traffic (flooding the network) in the direction of a server, thereby limiting users and necessary legitimate access to the server [7].

There are several common attacks methods known by the security community. They are divided into two main categories: Flood attacks and malformed packet attacks.

2.1 The Flood Attack

Flood attacks are quite common and they intend to saturate network in order to crash routers and switches or flood systems with more traffic than they can handle. Unfortunately, the tools required to mount such attacks are freely available on the internet and even malicious users with little or no experience can use them. The tools include, the Smurf Flood Attack, whereby, the attacker sends a small number of ICMP echo packets to a broadcast address that defines several hosts. The replies from all those hosts are sent simultaneously to the victims, exhausting all the available bandwidth and possibly processing power.

In Transmission Control Protocol (TCP) SYN attack, a malicious user will disregard the handshake protocol, the result is that the entry in the queue for pending connections is blocked until the timeout expires. If the malicious client sends a burst of such requests, it definitely reduces the throughput of the network and could cause an eventual shut down. Consider a typical 100MIPS server that can handle around 2000 connections per second [8], when the minimum standard TCP connection queue is 2048 slots [9] being subjected to this condition.

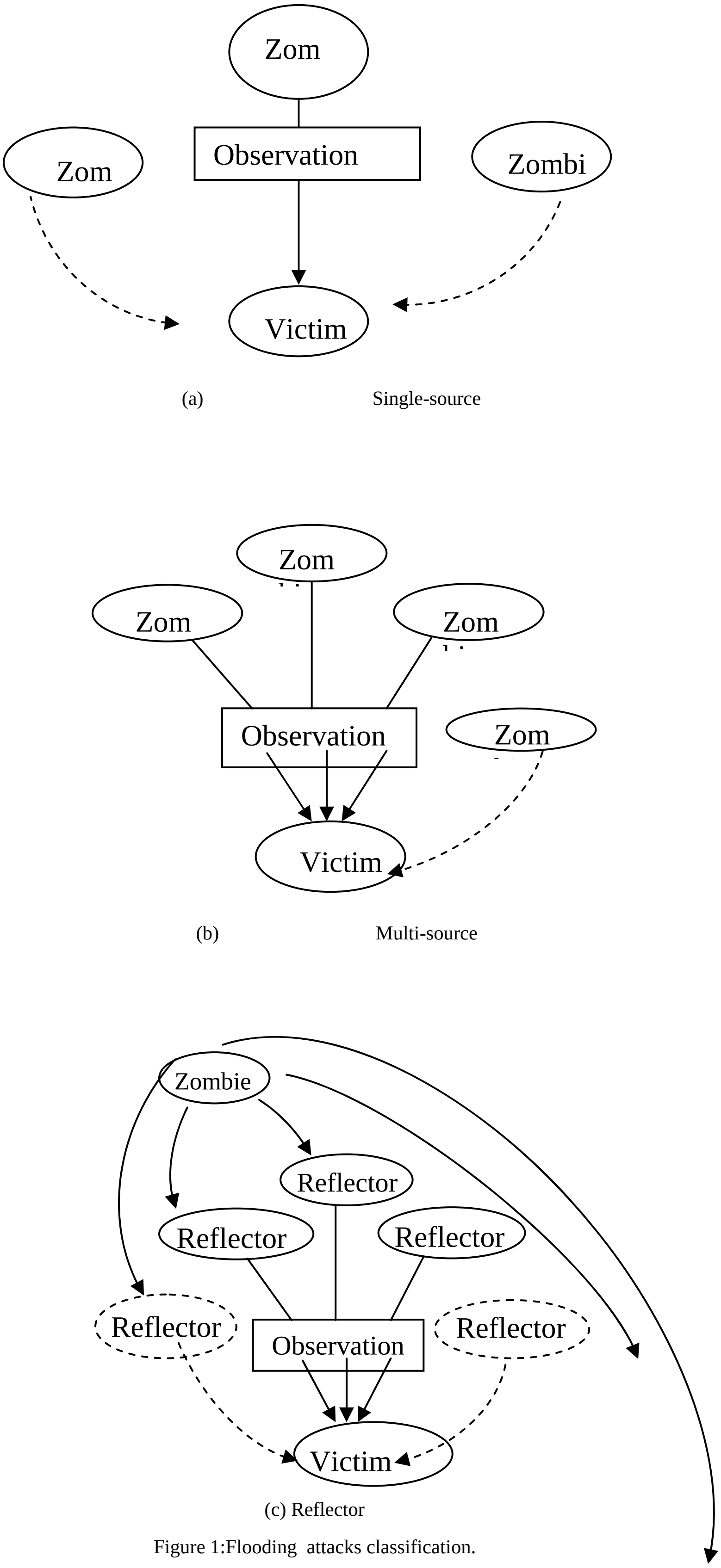
The UDP flood attack is possible because of the connectionless nature of the UDP protocol. Since no connection procedure is necessary, the attacker may send packets to random ports, causing the victim to allocate CPU cycles in order to determine which application listens to those ports. When it realizes that no application listens on the ports, the victim will generate a destination unreachable response ICMP packet and will send it to forged originating address. If enough packets are sent to the victim, the system may go down.

Another scenario is the ICMP flood attack, whereby an attacker sending a large number of echo ICMP packets to the victim. As the victim cannot keep up with the load, the system will experience performance depredation.

E-mail Browsing flood attack essentially, consists of sending a huge number of e-mails to the target in order to fill the storage space and/ or the bandwidth.

It is discussed in the literature [7] that with the exception of the UDP flood attack, the other attacks can be avoided by patching the operating system. The UDP attack is difficult to cope with since there may be several applications listening to ports and preventing access by means of firewalls. This may severally reduce functionality. Such attacks cannot be instigated unless we find a way to tell legitimate request from the fake ones. In addition to that, mitigating technology must not incur a significant overhead since this may open a new avenue of attacks.

Flooding attacks are also classified as (a) single – source (b) multi source, or (c) reflected based on the number of attacks and their location, with respect to the observation point and victim. This is shown in Fig 1 below.



(a) Single-source

(b) Multi-source

(c) Reflector

Figure 1:Flooding attacks classification.

2.2 The Malformed Packet Attack

The malformed packet attack is another wide-spread type of DoS attack. The purpose of this attack is to send ill formed packets to the host and take advantage of the bad design of the code that processes the packets. Effects range from unacceptable degradation of performance to system crashes.

There are several malformed packet attacks which include the Ping of Death Attack which involves sending an ICMP echo packet that is much larger than the maximum IP packet size of (64kbytes). At destination, since Transmission Control Protocol/ Internet Protocol (TCP/IP) implementation fail to reconstruct the packets, crashing or rebooting of the system occurs.

In the Chargen Attack, a variant of the UDP flood attack, the attacker uses the port 19 (chargen) of an intermediary system normally used as an amplifier. The attacker sends a forged UDP packet on port 19 of the intermediary system which in turn replies with a string of characters back to the victim who then sends back an echo of the string and the loop is created which subsequently rapidly exhausts the bandwidth between the victim and the intermediary system.

in teardrop Attack**, d**ue to poor implementation, some systems fail to correctly cope with packet fragment that have incorrect offset, making proper reassembly impossible. Instead of gracefully discarding the packets, the implementation in question simply reboots or halts the system.

The Land Attack **involves the** crashing or rebooting of a system, when it encounters a forged packet which contains the same address as both the origin and the destination.

The Win Nuke Attack is specifically targeted against windows machines to which attackers send out –of - band data to a specific port, causing the system to crash or reboot.

2.2 Other Variants of DoS

Another classification of the DoS attacks may be according to the number of parties involved in the attack:

Uni - source attacks – Involve one attacker that is targeting a single victim.

Multi- source attacks - several hosts referred to as zombies unwillingly participate as attackers, being compromised by the head of the operation. Although more difficult to put into practice, this type of attack is the most dangerous and most difficult to fight against. It is also known a Distributed denial of service (DDoS) attack.

3.0 **Framework for Classifying Denial of Service Attacks**

There appears to be a strong argument among researchers for an efficient classification framework. That is, a framework to classify DoS attacks into their appropriate categories, culminating to a valid identification framework. It has been argued in a literature [6] that this identification framework can be used as part of an automated DoS detection and response system.

The gap in research identified in the literature is work on a framework for detecting and identifying DoS attacks dynamically. It is obvious from network security literature that launching a Denial of Service (DoS) attack is trivial, but detection and response is a painfully slow and often a manual process. Automatic classification of attacks as single or multi-source then becomes necessary.[10]

It is noticeable from available literature [11] that there had been much research addressing automated classification of attacks.

This section describes a framework to classify attacks based on header analysis, ramp – up behavior and spectral analysis. It is however suggested in the literature that this three – pronged approach is necessary to deal with an increasing level of difficulty in classifying attacks depending on the level of IP header spoofing present in the attack.

3.1 Classification Of Dos by Header Contents

Most attacks spoof the source address concealing the number of attackers. However, other header fields, such as the fragment identification field (FID) and time – to – live field (TTL), can be indirectly interpreted to provide limits regarding the number of attacks. Such techniques have been used before to identify multiple interfaces on routers [11] and count number of hosts behind a NAT box [12]. These techniques work because many operating systems sequentially increment the ID field of each successive packet generated. As a result, all packets generated by the same host will contain monotonically increasing ID values. In addition, assuming the routes remain relatively stable during the attack, the TTL values will remain constant for the same source – destination pair. Thus for attacks where the ID and TTL fields are not simultaneously forged, we use the algorithm outlined in figure 2.2 to estimate the number of attacks and classify attacks as single – or multi – source.

Some related literatures show that some researchers using this algorithm try to estimate the number of attackers by counting the number of distinct ID sequences present in the attack. Packets are classified as belonging to the same sequence if their ID values are separated by less than idgap and the TTL value remains constant for all packets.

It has been established [6] using this algorithm, high volume attacks exhibit ID values that typically wrap around within a second. Therefore using a small idgap also limits collision during sequence identification. If a packet does not belong to an existing sequence, it forms the beginning of a new sequence. In most cases, attack packets arrive close to each other and have an idgap of one. An attack sequence must consist of at least 100 packets to identify a distinct attacker.

3.2 Classification of DoS by Ramp – up Behavior

Ramp – up behavior is changes in the traffic volume of the attack as a function of time. In a multi – source attack, a master typically activates a large number of zombies by sending a trigger message that either activates the zombies immediately or at some later time. When observed near the victim, this distributed activation of zombie results iin a ramp – up of the attack intensity due to the variation in the path of latency between the master and the zombies and weak synchronization of local clocks at the zombies. In contrast, single – source attacks do not exhibit ramp – up behavior. Thus, the presence of a ramp – up provides a hint as to whether the attack is single – or multi – source. However this method cannot robustly identify single – source attacks since an intelligent attacker could create an artificial ramp – up from a single site.

3.3 Dos Classification by Spectral analysis

A more robust method of classifying attacks as a single – or multi – source is to consider their spectral characteristics. It was observed in some research writing that attack stream have markedly different spectral content that varies depending on the number of attackers. Very little research papers have described methodology for analyzing the spectral characteristics of an attack stream.

This requires treating the packet trace as a time series. A research paper, [6] divided the attacks stream into 30 second segments, defining x(t), $0 \leq t < 30,000$ as the number of attacks packet arrivals on each 1ms interval. In this paper, a linear least – square method was used to compute the slope of x(t) and verify that the difference between the slope and zero is statistically insignificant within a 95% confidence interval [6]. Further, as reported in the paper, the researchers conditioned x(t) by subtracting the mean arrival rate before proceeding with spectral analysis.

For stationary segments, the researchers compute the power spectral density by performing the discrete – time Fourier transform on the autocorrelation Function (ACF) of the attack stream. The autocorrelation of an attack stream is a measure of how similar the attack is to itself shifted in time by offset k [13], [14] When k = 0, the attack stream was compared to itself, and the autocorrelation was maximum and equal to the variance of the attack stream. When k > 0, the attack stream was compared with a version of itself, shifted by log k. The autocorrelation sequence r(k) at log k

$$C(K) = \frac{1}{N} \sum_{t=0}^{N-K} (x(t) - \bar{x})(x(t+K) - \bar{x}$$

……………………..(1)

r(k) = c (k) c(0) ……………………………(2)

Where x^{-} is the mean of x(t) and N is the length of attack stream x(t). The power spectrum S(f) of attack obtained by the discrete – time Fourier transform of the autocorrelation sequence of length M.

$$S(f) = \sum_{k=0}^{m} r(k) e^{N-K2\pi fK}$$ ………….……………………..(3)

The highest frequency observable by this procedure is 500Hz, if according to reviewed literature, 1ms intervals is considered and the Fourier transform is symmetric. Intuitively, the spectrum S(f) captures the power or strength of the attack stream contained at a particular frequency. [6] Posited that once the spectrum is generated, a technique is needed to compare the spectral characteristics of different attacks. Therefore, for each attacks the cumulative spectrum P(f) is defined as the amount of power in the range of 0 to f. P(f) was normalized by the total power to get the normalized cumulative spectrum(NCS). C(f) [14]. Finally the description in [12] defined the quartile F(P) as the frequency at which the NCS captures P percent of the power. Formally:

$P(f) = {}^{f-1}\sum_{i=0} (s(i) + s(i+1)/2$……………….….(4);

$C(f) = P(f)/ p(F_{max})$……….………………….(5);

$F(P) = \min f$ such that $c(f) \geq 0$

$0 \leq f \leq F_{max}$

F(p) was used as a numerical method of comparing power spectral graphs.

It can be inferred from related research work that the key insight here is that multi source attacks shift spectrum to lower frequencies.

Research by Alefiya Hussain [6] made quantification by picking quartile of 60% value of attacks. His observations indicate single – source attacks have a linear cumulative spectrum due to dominant frequencies spread across the spectrum. This causes F (60%) to be in the range of 240 - 296Hz. In contrast, multi source attacks have localization of power in lower frequencies resulting to F (60%) in the range of 142 - 210Hz.

Although in the research work [6], the researcher used a 60% quartile, that choice is somewhat arbitrary. However, the important characteristic is that it captured the trend in frequency distribution of the spectra.

```
Let P = {attack packets}, Pi CP, P = $U^{n}_{i=2} P_{i}$
if ∀ pϵp
        ID value increases monotonically and
        TTL value remains constant
then single – source
Elseif ∀ pϵ$P_{i}$
        ID value increases monotonically and
        TTL value remains constant
then multi – source with n attackers else unclassified
```

Figure 2.2: Pseudo code to identify number of attacks on the header content

3.4 The DoS Network Security Framework: The Intrusion Detection System (IDS)

In the discussion of any complex subject such as network security, a framework would help to ease our understanding as it guides our conceptualization of the subject matter.

It is found in the bulk of the literature in security architecture that the intrusion detection system forms the core of the discussion on security framework. Hence it was deemed necessary for the effective exploration of this subject to review relevant literature relating to intrusion detection system (IDS) with the view to articulating a general framework for an intrusion detection system adapted for network defense from DoS.

Authors are of the view that intrusion detection techniques employed to detect attacks are now not new. However, there seems to be some agreement among authors that intrusion detection until recently has been employed for perimeter security and detecting denial of service attacks.

It can be said in the literature that researchers do not differ much on the goal of intrusion detection: IDS are a system which automatically monitors and analyses the events occurring in the computer or network and identify manifestation of attacks.

Intrusion detection systems (IDS) can be classified based on two concepts; matching of the previously seen and hence known anomalous patterns from an internal database of signatures or building profiles based on normal data and detecting deviation from the expected behaviors. The first approach is referred to as misuse detection and leads to signature Based IDS while the second is Anomaly Detection and leads to Behavior based IDS. The signature based systems though have very high detection accuracy but they fail when an attack is previously unseen. On the other hand, Behavior based IDS may have the ability to detect new unseen attacks but have the problem of low detection accuracy [15] [16] [17].

Also based on the mode of deployment, the intrusion detection systems are classified as Network based, Host based, and Application based.

Network based systems make a decision by analyzing the network logs and packet headers from the incoming and outgoing packet since they are deployed at the periphery of the network. Though they are easy to manage and give a centralized control, they have to work with limited information and are further constrained in case of encryption and network address translation. Host based systems monitor individual systems and uses system logs extensively to make any decision.

Fred Cohen published in 1984 that detection of computer viruses is undecidable and NP-hard [18]. In laymen’s terms, this means that it is impossible to detect every type of an intrusion in every type of case, and that the resources needed to detect intrusions grow with the amount of network traffic.

Paul Helman, et al in 1992 used a scale of 0 to 1 to represent normal behavior (0) to misuse (1) [19]. The purpose of an intrusion detection system is to provide the rating for computer activities. Helman showed that problems facing this scheme include imperfect and incomplete information, plus the large number of potential events which is estimated to $10^{100.}$ When groupings are done to reduce the number of possible events, this becomes an NP-Hard problem to reduce singleton groups. Helman calls the above a modeling approach. An alternative involves non-modeling approaches which include heuristics, clustering algorithm, and statistics.

Data mining approaches, clustering, naïve Bayesian classifiers, Bayesian networks, hidden Markov models, decision trees, artificial neural networks, support vector machines, genetic algorithm, agent based approaches and many others have been described in order to detect intrusion.

Data mining based approaches for intrusion detection are based on building classifiers based on discovering relevant patterns of program and user behavior. Association rules [20] and frequent episodes are used to learn the record patterns that describe user behaviour [21] [22].

Data mining approaches can deal with symbolic data and the features can be defined from packets and connection details. Thus, mining of features is limited to entry level of the packet and also require the number of attributes to be large and the records are sparely populated, otherwise they tend to produce very large number of rules which increase the complexity [23]. Clustering of data has been applied extensively for intrusion detection using various clustering methods including K-means, Fuzzy c-means many others [24], [25]. However, one of the main drawbacks of clustering techniques is that it is based on calculating the distance between the observations and hence the attributes of the observations must be numeric. Symbolic attributes can not be used for clustering which result in inaccuracy.

The DoS security breaches have a number of proposed approaches to their remedy, which include completely eliminating the attack, mitigating effects of the attack and discouraging the attacker itself. It is recommended in various papers on the subject that these approaches may not yield optimal results when used individually hence the need for a solution with a view to having an approach complemented by another one or two.

3.5 The Requirements Of an Improved IDS Requirements

In [26], authors have defined a set of desirable characteristics for IDS by focusing on two themes: Functional and performance requirements.

The functional requirements require that an

- IDS must continuously monitor and report intrusion,
- IDS should have a very low false alarm rate,
- IDS should provide enough information to repair the system in the case of detection of intrusion notice that these requirements lean on basic IDS goals. In fact, conventional IDS solutions focus only on alerting administrators without suggesting any corrective actions.
- IDS must detect and react to distributed and coordinated attacks. This detection feature is one of the most difficult because it needs a huge distributed amount of information in addition to the hard task of synchronization between different hosts.
- The IDS should be adaptive to network topology and configuration changes [26].

Also, the performance requirements include:

- Intrusion should be detected in real – time as it should be reported immediately in order to minimize network damage.
- The IDS must be scalable in order to handle additional computational and communication loads.

In addition to the fore mentioned shortcomings, it is posited in referenced literature that the adaptive intelligent agents' solutions is aimed at overcoming the following limitations:

- Many of the existing network and host based IDS perform data collection and data analysis essentially by using a monolithic architecture [27]. The centralized detection scheme suffers from a number of problems:
    i. A control analyzer presents a favorable target to attackers. If an intruder manages to decapitate it, the entire network loses protection,
    ii. A high network load leads to excessive data traffic, system suffers from scalability problems. A single analyzer unit limits the network size,
    iii. Since network data collection is performed in a host different from the one in which analysis is performed, intruders can perform insertion and evasion attacks [28].

Intrusion can be conducted through several steps that occur at different hosts, and consequently cannot be detected by a single sensor. The cooperation of different sensors becomes an imperative for the identification of distributed intrusion. Thus, intelligent agents offer a new approach to IDS intrusion detection implementation. Typical adaptive intelligent agents' technology can solve the set of the shortcomings mentioned above.

4.0 The Intelligent Agent Concept and application to a new model of DoS system

The intelligent agent is a growing area of research and new application development. Having highlighted the main requirement for IDS, the intelligent agent concept seems to be a candidate approach to fulfill these requirements. What is the intelligent agent concept? In the referenced literature for this chapter, until now, there is no internationally accepted definition of an intelligent agent concept [29]. The term agent is a concept used in different areas and having different meaning depending on the context [30]. Nevertheless, different types of agents reflect a set of properties, which common among them [31] are described as follows:

a. Autonomy: This is the ability of an agent to operate without direct intervention of humans or other agents and have some kind of control based on its internal and/or external environments.

b. Co-operation: An agent is co-operative and is able to have a social ability. This sociability allows an agent to interact with other agents for the purpose of performing tasks that are beyond the capability of a particular agent. This capability goes from delegation (distribution of sub – tasks) to peer - to – peer inter working.

C. Proactive: It is the agent ability to anticipate situations and change its course of action to avoid them. Proactive agents are capable of exhibiting goal – direct behaviors by taking some initiative [32][33]

d. Reactivity: This kind of behavior means that the agents react on real- time to changes that occur in its environments.

e. Adaptability: Is the ability of an agent to modify its behavior over time to fulfill its problem – solving goal.

f. Intelligence: The term “intelligence” means that the agent is able to exhibit a certain level of intelligence priority, ranging from predefined actions (planning) up to self learning (define new action).

g. Flexibility: Is the ability of an agent to adapt itself so as to cope with the environment in which it is situated.

h. Mobility: An agent is mobile. It is capable of moving from one location to another in order to perform a particular task or to react to particular event.

To model intrusion detection, the intelligent agent must combine the cognitive (knowledge – based) to reason about complex attacks with reactive capabilities (stimulus response) to react to the environments changes. Thus, an agent has three functions: an event filtering function, an interaction function, and a deliberation function.

A security event is characterized by its type, its observation point, as well as attributes. Here, the event filtering function filters security events produced in the network, according to event classes specified in a detection goal. The filtered events are then stored, waiting to be treated by the deliberation function.

The interaction function describes interaction and it allows them to communicate their analyses and knowledge and mental attitudes (beliefs, suspicion). In fact, manager agents interact with local agents by: - sending goals derived from security polices, delegating specific functions of monitoring/ detection and specifying the various domains to monitor,

Considering the unpredictable character of the agent environment behavior; the computer network, most researchers adopted BDI solution [39][40] for modeling the security management system. With the deliberation function the agent is able to reason and extrapolate by relying on its mental attitudes, built knowledge and experience, in such a rational way as to finding the adapted answer. The agent uses its beliefs resulting from the filtered events and beliefs of the neighboring agents for reaching its specified goals. When a goal is reached (an attack is detected), it executes the appropriate actions. This is shown in Fig 2 below.

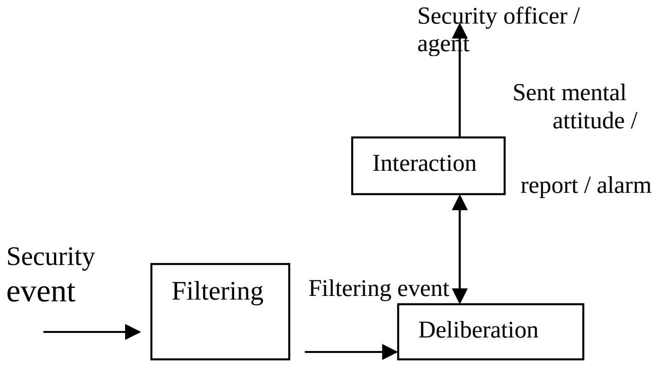


Figure 2. Interaction between Agent and Function

4.1 The DIANA Agent Architecture

The DIANA architecture [35] is the shell of the proposed intelligent agent strategy for network protection as proposed by this work, The DIANA agent architecture consists of two main components: the brain, which is responsible for managing agent skills and skills, which provide the agent with capabilities and behaviors.

The brain (figure 3) offers two types of necessary facilities for the agent operations: local and inter agent facilities.

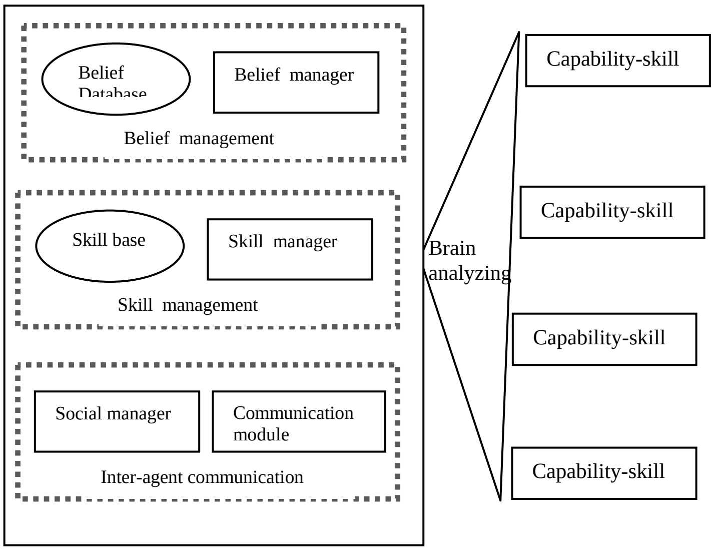


Figure 3: DIANA Agent Architecture

The main role of the brain is to manage both agent’s belief database and agent’s skill base. “An agent belief expresses its expectation about the current state of the world and about the likelihood of a course of action achieving certain effects”[34]. Beliefs hold network management information as well as information about the agent itself and the other agents. These beliefs can be accessed concurrently by several skills, therefore, the Belief Manager maintain the integrity and coherent access to the Belief Database.

Skills can be downloaded dynamically into the agent inside its skill base. The main role of the skill manager is to check the availability of pre-requisite skills required by newly loaded skills and if they not yet loaded, it must search for them either locally or on distant agents. It is also responsible for the disposal of un-useful skills to keep the agents size as small as possible. During its operations, the skill can update or delete existing beliefs or create new ones. A skill operation may depend on belief created by other skills, and the skill manager is therefore in charge dispatching asynchronously these beliefs to the interested skills in a transparent way. It holds all the necessary information about the skill in the base.

The brain Analyzer is responsible for the parsing of the messages that the brain receives, either from the skill or from the inter-agent communication.

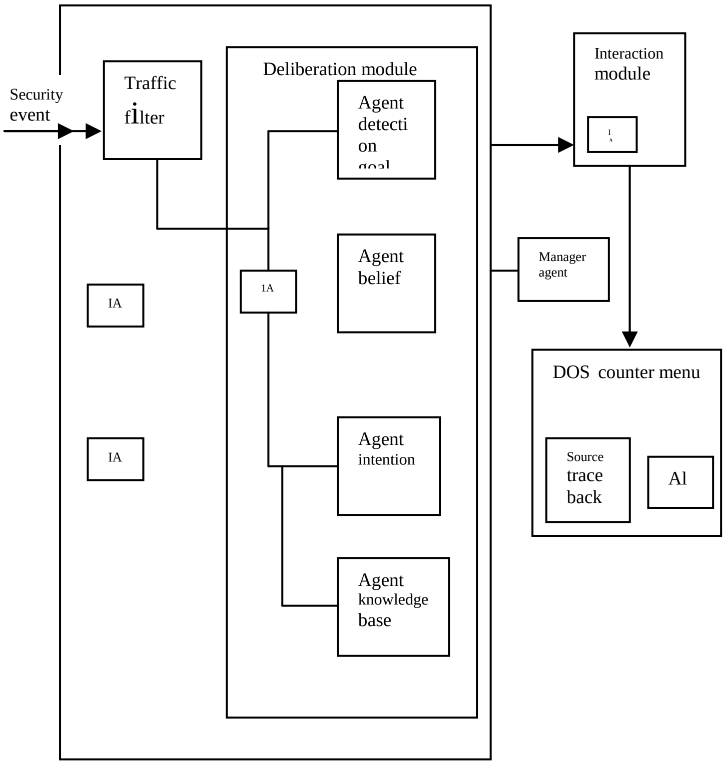
Both the communication module, which is responsible for managing interaction with the other agents and the social manager, which holds information about the other agents, support inter-agent communication facilities to the agent.

4.2 The proposed Intelligent Agent Intrusion Detection System

The idea of distributing the intrusion detection system using agent software is not entirely new. However, most of the related works emphasize static agents instead of mobile one. Applying mobile agent technology to IDS has been carried out within only a few research projects.

In 1999, a project at the information-Technology Promotion Agency (IPA) in Japan involved an intrusion Detection Agent (IDA) system [36]. IDA is a classic hot-based system that relies on mobile agents mainly to trace intruder among the various hosts involved in an intrusion. In the same year the project MICAEL [37] pursued a more ambitious aim where the entire system is based on adaptive intelligent agents. Nevertheless, only the architecture description has been presented and no details have followed so far. In 2000, an IDS framework based on mobile intelligent agents has been described in [37]. Unfortunately, detection is dealt with superficially. In 2002, [38] describes an IDS designed as a mobile application that roams the network to detect attacks and track intruder. IMA-IDS are a distributed intrusion detection system using mobile and intelligent agents.

It is too tedious to detect a malicious action through the network, especially when considering multiplier distributed events which invariably are equally simultaneous. Most of the current IDS assume that the environment is static, whereas in reality the environment is dynamic and unpredictable. So, to detect without mistakes, an intrusion, subsequently make the appropriate decision and at an optimal time will require a cooperation of different sensors and analyzers. Thus, the need for an adaptive intelligent agent, with the attendant solution the problems of current IDS.



IA ≡ Intelligent agent

Figure 4 A Block diagram overview of the proposed system

The deliberation module is a major component of the multi agent system. The deliberation module interacts directly with information resources mainly different specialized database entries. The databases are: The agent detection goals, Agent beliefs, Agent intention, and Agent knowledge base. The intelligent agents do not interact directly with these databases but does so through the deliberation module as depicted in fig. 3.4. Agents access and update these databases as the multi-agent system runs.

In the block diagram (fig. 4), the traffic filter is a component of the system that filters incoming traffic. This component constitutes the filtering function of the intelligent agents. This filtering function access the detection goal through the deliberation module. The Agent detection goal is a database of security event classes. A security event is only collected when the event matches the event classes specified by the detection goals.

The agent belief is a collection of filtered events. What we can call likely attacks, or possible attacks.

The deliberation module which is also a function of the intelligent agent, test if a belief matches an attack. If it matches, then a detection goal is reached and a list of intentions is sent to the interaction module. The agent intention is a database of security countermeasures (actions carried out). The interaction module is a component as well as a function of intelligent agent that executes it security counter-measures required to protect the network resource of the Denial of Service attack.

An intelligent agent interacts with other intelligent agents to get specialized help or other functionality as shown by the IA symbol.

The manager performs some sort of co-ordination function. In doing this it also interacts with the agent knowledge base to get information about the types and specialized functions, about other agents within the multi-agent system. It coordinates inter-agent interactions. The structure of interaction with other specialized agents is depicted in the fig. 4

This collector agent will be interested in a set of event categories.

Three other agents are also required:

Correlator agent: this agent will carry specific information, called critical and send it to another agent called the analyzer agent.

Analyzer agent: analyzer agent are the engine of this proposed agent system. Several kinds of analysis such as classical signature detection, anomaly detection is integrated in this agent.

Manager agent: this agent gathers collected information and distributes it to analyzer agents.

5.0 RESULTS

The model derived (Fig 4) was systematically generated from the basic requirements of remedying DoS attacks, and which was leveraged on the system development scheme of an adaptive intelligent system. It thus becomes imperative that the system developed there from is imbued with the inherent characteristics of the adaptive intelligent agent.

6.0 CONCLUSIONS

Having reviewed the properties of intelligent agents as presented in referenced literature [31], it can be concluded that intelligent agents provide a more coherent and flexible approach to network intrusion detection and security management. The security management architecture based on the concept of intelligent agents can be conceived as if it were made of the autonomous intelligent agents co-operating with each other to achieve Global Security Policy.

According to the attributes described above, referenced literature [32], [33] argue for the use of intelligent agents to improve the characteristics of the existing IDS and to overcome the limitations described in this work

a. Reducing Network Load: Existing IDS are faced with the problem of performing a huge amount of data over transfer. Abstracted forms of this data are usually sent from all locations in the network to the central site in order to e processed. Sending a huge amount of data causes an increase of network loads. Intelligent Agents offer the opportunity to overcome this problem by eliminating the need of so much data transfer. The processing program (agent) can be dispatched to the host containing crucial data. This will reduce network traffic since an agent is smaller than the processed data.

b. Overcoming Network Latency: Intelligent Agents are able to dispatch from a host to carry out operations directly to the remote point of interest, thus agents scans provide an appropriate response faster than hierarchical IDS that has to communicate with a central coordinator based elsewhere on the network.

c. Asynchronous Execution and Autonomy: Agents can be stopped and started without distributing the rest of the IDS. Notice that the intelligent agents are able to continue to operate autonomously even if the host platform where it was created is not available or is disconnected from the network. Intelligent agent framework provides IDS with the possibility of continuing to work even when a central controller is down.

d. Dynamic Adaptation: Intelligent agents can be retracted, cloned, dispatched, killed or put to sleep as network's configuration, topology and traffic characteristic change over time. As the number of nodes in the network increases, agents can be cloned and dispatched to these new elements.

Referenced literature [34] indicated the need for intelligent agents to possess some form of capabilities and interactions in order to model intrusion detection.

REFERENCES

[1] P. Ramasubramanian and A. Kannan, "Intelligent Multi Agent Based Multivariate Statistical Framwork for Database Intrusion prevention system". School of Computer Science and Engineering, Anna University, India.

[2] Cisco System, Inc, "Core element of the Cisco Self – defending network strategy". White paper 1992 – 2005.

[3] Todd Lammle, " Cisco Certified Network Associate Study Guide Sixth Edition". Chapter 10 page 611 -613, 2007.

[4] Valer Bocan, "Threshold puzzle: The evolution of DoS- resistant authentication". PERIODICAL POLITEHNICA, Transaction or automatic control and computer science vol 49 2004.

[5] Valar Bocan, "Development in Dos Research and Mitigating Technologies:. Department of Computer Science and Engineering, Politehnica University of Timiora, Bd.V. Parvan, 300223 Timisoara, Romannia.

[6] Aleifa Hassan John Haidiman Christos Papadopoulos, "A framework for classifier denial of service attack". USC / Information science institute. Jan 2007.

[7] Ping – Herng Denny Lin, "Survey of Denial of service counter measures". California State University Fullerton,

[8] November 2000.

[9] Oliver Spatscheck and Larry Perterson, "defending against denial of service in scout, in proceedings of 3rd USENIX / ACM symposium on OSDI, 88. 59-72, Feb 1999.

[10] Prashant Dewan, Parths Dasgopts, Vijay Karamcheti ,"Defending Aganist Denial of service Attacks Using Secure Name Resolution".

[11] Nail Spring, Ratul Mahajan, and David Watherall, "Measuring ISP topologies with rocket fuel". In proceeding of ACM SIGCOM, 02 August 2002.

[12] Steven Bellorin, “A technique for counting noted hosts”. In proceeding of the ACM SIGCOMM Internet measurement workshop November 2002.

[13] Genge Box, Gwilyn Jenkins, and Gregory Reinsel, “Time series analysis forecasting and control”. Prentice-Hall, New york, NY, 1986.

[14] Ronald Bracewell, “The Fourier Transform and Its Applications”. Series in electrical engineering. Mc Graw Hill. New York NY, 1986

[15] H. Debar, M. Becker, and D. Sibon, “A neural network component for an intrusion detection system”. In proceeding of the IEEE computer society symposium on research in Security and privacy, 1992.

[16] S. Kumar and E.H. Spafford, “An application of pattern matching in intrusion detection”. In technical report CSDTR-94-013, Purdue university, 1994.

[17] K. Ghosh, “Learning program behaviour profiles for intrusion detection”. In proceedings of the 1st USENIX workshop on intrusion detection and network monitoring, 1991.

[18] Cohen, Fred, “ Computer viruses: Theory and experiments,” 7th DOD / NBS computer security conference, Gaithersbuy, MD, September 24-16 2004.

[19] Helman, Paul, Liepins, Gunar, and Richords, Wynette, “Foundation of intrusion detection”. The IEEE computer security foundation workshop V. 1992.

[20] R. Agrawal, T. Imielinski, and A. Swami, “Mining association rules between set of items in large database”. In proceeding of the 1993 international conference on management of Data (SIGMOD93), ACM press, vol(22) issue 2, 1993, page 207-216.

[21] W. Lee, and S. Stolfo “Data mining approaches for intrusion detection”. In proceeding of the 7th USENIX security symposium, 1998.

[22] W. Lee, S. Stolfo, and K. MOK. “Mining audit data to build intrusion detection models”. In proceeding of the 4th international conference and data mining. AAA1 press, 1998.

[23] http://www.sans.org/resources/idfaq/hostbase.php.online article (last accessed: July 06 2006).

[24] L. Portnov, E. Eskin, and S. Stolfo, “Intrusion detection using neural network, and support rector machines”. In proceeding of the 2002 international joint conference on neural networks. IJCNN’ 02; IEEE press.

[25] H. Ryan, M.J.Lin, and R. Miikkullainen. “Intrusion detection with neural networks: In advance in neural information proceeding system”. MIT press, 1998.

[26] W. Jensen, P. Mell, T. Karygiannis, and D. Marks,” Applying mobile agents to intrusion detection and response”. Technical report, NIST interim report.

[27] J.S. Balasubramaniyam, J.O Garlia. Fernadas, D. Isacoff, E. Spafford, and D. Zamboni, “An architecture for intrusion detection using autonomous agents”. In proceedings of the 14th Annual computer Security Applications conference, 1998.

[28] T.H. Ptalek and T. N Newsham. “Intrusion evasion, and denial of service: Eluding network intrusion detection”, Technical report, secure Network Inc. Jan 1998.

[29] H.S Nwana and M. Wooldridge, “Software Agent Technologies”. B. T Technology journal 1996.

[30] S. Corley and al, “The Application of intelligent Agent m to Network and service management”, 5th IS $ N conference, Antwerp, Belgium, 25- 28 May 1996.

[31] R. Oliveira, “Network management with knowledge of Requirement: Use of software Agents”. PhD thesis, 1998.

[32] M. Wooldridge and N. R. Jennirys “Intelligent Agents: Theory and practice”. Knowledge Engineering Review, 1995.

[33] H. Labroid,”Error control in wireless ATM networks”. Thesis 1998.

[34] J. P.Muller,” The Design of intelligent Agents- A layered approach”. LNA1 state-of-the art survey, springer, Berlin, Germany, 1996.

[35] M. Asaka, S. Okasa wa, A. Taguchi, and S. “ A method of tracing intruders by use of mobile agents”. In INET’99, 1999.

[36] De Ouerinz, L.F.R Da Costa, and L. Primez, Miceal: An autonomous mobile agent system to protect new generation networked application. In 2nd Annual workshop on recent advances in intrusion in detection 1999.

[37] M.C. Bernades and E. Dos Santos Mareira. Implementation of an intrusion detection system based in mobile agents. In international symposium on software engineering for parallel and distributed system, 2000.

[38] Trapethi, T. Ahmed, S patlak, M. Carney, M. Koka, and P. Dokas. Active monitoring of network system using mobile agents. Technical report, university of Minnesota, 2002.

[39] A.S. Rao and M.P. Geogeff, “ Modeling rational agent within a BDI. Architecture” Technical note 1991.

[40] A.Rao and M.Geogeff, “BDI agent: from theory of practice technical note, 1991.

[41] Ping-Herng Lin, “survey of denial of service counter measures”, California state university Fullerton November 2000.

[42] Bruce Scheier, “Distributed denial of service attacker” Cryptogram newsletter, Feb 2000. (http://www.Counterpane.com/cryptogram_0002.html#DistributedDenial-of-serviceAttacks)

[43] P. Ferguson, D. Senie, “network Ingress filtering: Defeating denial of service attack which employ IP source address spooling”, RFC 2267, Jan 1998.

[44] Neil’s Ferguson, Bruce Schenier, “Practice cryptogram”, Wiley publishing, Inc. 2003.

[45] DongGrok Park, JungJorn Kim, Colin Boyd, Ed Dawson, “Cryptographic salt: a counter measure against denial-off-service attacks”

[46] Tuomas Aura, Pekka Nikandar, and Jussipkka Leiwo. “DoS- resistant authentication with client puzzle”. In proceedings of Cambridge security protocols workshop 2000, LNCS, Cambridge, UK, April 2000. Springer-Verlas.

[47] Shon Harns, “Dos defense”. Information security magazine, Sept 2001.

[48] Ari Juels and John Brainerd, “Client puzzle: Cryptographic defense against connection depletion attacks”. In S.Kent, editor, proceedings of NDSS, as, 1999.

[49] Marcel Dekker. Security of internet. The Froehlich / Kent Encyclopedia of telecommunication vol. 15 1997.

[50] Roger M. Needlam, “Denial of service: An example”, Communication of the ACM, vol 37, No 11. pg 42-46 Nov 1994.

[51] Progress Pressman R.D, “Software Requirement: An Practioners Approach”, Fifth Edition Mc Graw – Hill. Service in computer science http:// www.mhhe . com

[52] Jocobson, I, Booch, G. and J. Rumbanyh, “The Unified Software Development Process, Addison –Wesley, 1999.

AUTHOR PROFILE

Engr. (Mrs) Ogechi Ihekweaba is a lecturer in the Department of Computer Engineering, Michael Okpara University Of Agriculture, Umudike, Abia State, Nigeria.

She is an ex student of Ido-Ani secondary School, Ondo State. Holds a Bachelor’s degree (B.Eng) in Computer Science & Engineering, a Master’s degree (M.Eng) in Computer Science & Engineering and she is at the verge of completing a Doctorate degree (PhD) in Computer Engineering.

Her area of specialization is Network Security and Computational Intelligence. She served as an Engineer with a Telecommunication outfit, CSAT COM Ltd. As a lecturer, she headed the Computer Science Department of OSISATECH. She is currently the SIWES and Seminar Coordinator for the Department of Computer Engineering, Michael Okpara University of Agriculture, Umudike.

She had several publications, and is a member of professional bodies which include: Nigeria Computer Society (NCS), Computer Professionals of Nigeria (CPN), Nigeria Society of Engineers (NSE). She is also a COREN registered Engineer.