

AFRICAN JOURNAL OF COMPUTING & ICT

Volume 11, Number 3, September 2018
https://afrjcict.net

ISSN 2006-1781

AFRICAN JOURNAL OF COMPUTING & ICT

Volume 11, Number 3, September 2018
https://afrjcict.net

CONTENTS

Editorial Board iv

From the Editor-in-Chief v-vi

A Survey of Drone Use for Socially Relevant Problems: Lessons from Africa 1-11
A. Nicki Washington
209 Thurmond Building, Winthrop University,
Rock Hill, SC 29733, USA

Design Evaluation of Academic Library Websites in Nigeria 12-32
Fasola P. Abifarin and Shaka A. Imavah
Department of Library and Information Technology,
Federal University of Technology,
Minna, Nigeria.

Development of a Web-based University Collaborative Tool for Effective Distributed Learning 33-43
Adeyinka Adedoyin[1], Onungwe J. Obariobe[2], A. S. Sanni[3] and Aduragba Tahir[4]
[1,2,3]Department of Information and Communication Science,
University of Ilorin, Ilorin, Nigeria.
[4]College of Engineering and Technology,
Kwara State University,
Ilorin, Kwara State, Nigeria

A Hybridized Semi-Global Sequence Alignment Algorithm for Intrusion- Detection Systems 44-70
Tinuke Omolewa Oladele[1], Olajide O. Adeola[2] and Jadesola Adejoke Emmanuel[3]
[1,2]Department of Computer Science,
Faculty of Communication and Information Sciences,
University of Ilorin, Ilorin, Nigeria.
[3]Department of Computer Science,
Bingham University, Abuja, Nigeria

Design and Implementation of a Multi-Layered Security Enterprise Resource Planning System for Mission Critical Applications 71-84
M. C. Nwafor[1], C. C. Okezie[2] and A. C. O. Azubogu[3]
[1]R & D Department,
Electronics Development Institute,
Awka, Nigeria.
[2,3]Electronics & Computer Engineering Department,
Nnamdi Azikiwe University,
Nigeria.

Electromyography Noise Suppression in Electrocardiogram Signal using Modified Garrote Threshold Shrinkage Function 85-94
Abdul Rasak Zubair[1], Yusuf Kola Ahmed[2] and Kareem Adeyemi Akande[3]
[1]Department of Electrical/Electronic Engineering,
University of Ibadan,
Ibadan, Oyo State, Nigeria.
[2,3]Department of Biomedical Engineering,
University of Ilorin,
Ilorin, Kwara State, Nigeria.

Tributes to Dr. John W. Mckee (1926-2009) of California and Dr. C. Clifton ('Cliff') Chancey (1955-2013) of Iowa, USA 95-103
Bamidele ('Dele) Oluwade
Dewade Systems Consult,
Nigeria

Call for Papers & Subscription Details 104

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

A Hybridized Semi-Global Sequence Alignment Algorithm for Intrusion-Detection Systems

Tinuke Omolewa Oladele[1], Olajide O. Adeola[2] and Jadesola Adejoke Emmanuel[3]
[1,2]Department of Computer Science,
Faculty of Communication and Information Sciences,
University of Ilorin, Ilorin. Nigeria.
Email: [1]oladele.to@unilorin.edu.ng, [2]ooadeola@gmail.com
&
[3]Department of Computer Science,
Bingham University, Abuja, Nigeria.
Email: [3]emmanueladejoke@yahoo.co.uk

ABSTRACT
In modern day computing, with persistent remote terminal access, transmissions and virtual networks, conventional methods find it difficult to provide adequate security for the data, information or the service. This challenge deserves new mechanisms to guarantee that the data or information being sent is well secured. A correct system will take into cognizance the protection of the data it processes and the information it stores and sends out at due time. This research work solves the problem of masquerade intrusion into data or information kept on the computer system by reducing the false positive alarm and false negative alarm which increases the true positive proportionately. The Semi - Global Alignment Algorithm which is a hybrid of Smith Waterman Alignment Algorithm is regarded as a dynamic and efficient technique in finding out the attacks. The reports of high false positive rate and false negative rate in the Semi-Global Algorithm have not been reduced to an acceptable level. This study therefore, proposes an improvement on the Semi - Global alignment Algorithm to develop a masquerade intrusion detection system. Experiment was conducted and the performance of the new algorithm was evaluated using Schonlau Et Al (SEA) dataset. The results give hit rate of about 90% and false positive, which show that the improvement on Semi - Global Alignment Algorithm yields increase in the percentage of hit rate of the matching and reduces the value of false positive rate as well as false negative of the alignment.

Keywords: *Hybridized-Cross-Semi-Global Alignment Algorithm, Masquerading, Sequence Alignment, Semi-Global Alignment Algorithm*

African Journal of Computing & ICT Reference Format:

Tinuke Omolewa Oladele, Olajide O. Adeola and Jadesola Adejoke Emmanuel (2018), A Hybridized Semi-Global Sequence Alignment Algorithm for Intrusion-Detection Systems, *Afr. J. Comp. & ICT*, Vol.11, No.3, pp. 44 - 70.

© Afr. J. Comp. ICT, September 2018; ISSN 2006-1781

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

I. INTRODUCTION

A controlled privilege for access to computer rooms is adopted by most organizations or personal computer users. Knowing fully that information system can be grouped into the hardware facilities and the information that are stored on the computer. Most information systems give adequate protection to their hardware component more than the protection they give their data and information, per adventure the data is more valuable than the worth of the hardware. In modern day computing, with persistent remote terminal access, transmissions, and virtual, and physical measures find it difficult to provide adequate security for the data, information or the service; the security of the hardware is assured to a large extent. In some instances, the computer to be protected is connected to a network and most of the threats arise from the network. Trent Jaeger says "computer facilities have been physically protected for three reasons; to prevent theft of or damage to the hardware, to prevent theft of or damage to the information and to prevent disruption of service". Most computer crimes are in fact committed by insiders, and most of the studies in computer security since 1970 have been pointed at the insider problem [1]. In spite of significant advances in the modern computer security in recent years, information in computers is more vulnerable than ever. Each major technological advancement in computing raises new security threats that require new security solutions, and technology moves faster than the rate at which such solutions will be developed. Why is a developed system that has capabilities of storing billions of information unable to protect that information? If the hardware component of the dependable system is secured, why not the software component especially the operating system? It implies it is far easier to develop a secured system than to develop a correct system. A correct system will take into cognizance the protection of the data it is processing and the information it stores and sends out at due time. [1] said that the publicity about computer security in the press, computer and software vendors have rarely taken the trouble to incorporate meaningful security measures into their systems.

Computers are widely available in every place and they are exposed to threats with the evolution and popularity of the internet. It was recorded that the terms threat and attack are commonly used interchangeably. A threat is a potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm. That is, a threat is a possible danger that might exploit vulnerability while an Attack is an assault on system security that derives from an intelligent threat; that is, an intelligent act that is a deliberate attempt (especially in the sense of a method or technique) to evade security services and violate the security policy of a system. Security attack can be classified as Active and Passive. An active attack attempts to alter system resources or affect their operation. Active attacks involve some modification of the data stream or the creation of a false stream and can be subdivided into four categories: masquerade, replay, modification of messages, and denial of service. On the other hand, passive attack attempts to learn or make use of information from the system but does not affect system resources that is, eavesdropping on, or monitoring of, transmissions. Two types of passive attacks are; release of message contents and traffic analysis [2].

This paper is based on masquerade attack detection on a computer system. Many researchers have proposed many approaches to develop a secured system that will check the alignment of commands of the legitimate user and an attacker. The confusion metrics that were used are Hit rate and false positive rate for the alignments of the two set of commands. The metrics need improvement to have a robust security system. This study will improve on this, by increasing the hit rate and reducing the false positive results found in the literatures. The concept of Bioinformatics sequence alignment algorithm will be used to improve on the existing system [3]. A Hybridized-Cross-Semi-Global Alignment and Semi-Global Alignment Algorithm was used to develop an intrusion detection system for identifying Masquerade attacks on computers.

II. LITERATURE REVIEW

2.1 Masquerade Attack

In the field of computer security, masquerading is one of the possible activities of unauthorized users, in which an attacker takes the identity of a rightful user in a computer system [3]. "Masquerade attacks are common security problems that are consequences of identity theft" [4]. "The masquerade attack is a class of attacks, in which a user of a system illegitimately poses as, or assumes the identity of another legitimate user" [4]. Identity theft in financial transaction systems is perhaps the best known example of this type of attack. Intrusions occur when an attacker get access to a legitimate user's login credentials such as username and password or when the terminals of users are opened to unauthorized users for use for which

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

there is no locking mechanism in place. Detection of this type of security breach is very hard to come by especially in a very complex system at initial stage, since the intruder had already assumed the behaviours or responsibilities of the legitimate user based on the valid credentials presented. "This difficulty underlines the importance of equipping computer systems with the ability to distinguish masquerading attacker actions from legitimate user activities" [3].

[5] points out that intrusion attacks are divided into eight basic categories:

i. Eavesdropping and packet sniffing (passive interception of network traffic);
ii. Snooping and downloading;
iii. Tampering or data diddling (unauthorized changes to data or records);
iv. Spoofing (impersonating other users, for example., by forging the originating e-mail address, or by gaining password access);
v. Jamming or flooding (overwhelming a system's resources, for example, by an e-mail flood or HTTP requests);
vi. Injecting malicious code such as viruses and Trojan horses (via floppy disks or e-mail attachments);
vii. Exploiting design or implementation flaws (often buffer overflows, which overwrite other data and can be used to get control over a system);
viii. Cracking passwords and keys.

Furthermore, masquerade attack, was second in the top five lists of electronic crimes perpetrated by outsiders after virus, worms and other malicious code attacks [6]. Machine learning (ML) algorithms have been applied to tackle this type of attack in this serious research field. They produce classifiers which can identify suspicious behaviours that may indicate an intrusion.

Considering the difference in the approaches shown above, it is glaring that Intrusion Detection System (IDS) has many shortcomings. These shortcomings call for improvement in these techniques so as to have a substantive solution for information system. Applying Bioinformatics pattern matching algorithms to complement existing traditional masquerade detection tool will be a major step in saving the society from the mess. Semi Global Positioning Algorithm has been used for masquerade attacks detection and it has proven to be an efficient algorithm till now. The challenge with Semi Global Positioning Algorithm is determining the best marking system [8].

2.2 Sequence Alignment Sequence alignment is widely studied and it is used to compare and analyze similarity between sequences. It has been frequently applied in the comparison of genomic sequences Protein Sequences" [3].[9] points out that: Most people share very similar gene sequences, while some regions of DNA sequences vary from one person to another with high frequency. Comparing variation in these regions allows scientists to answer the question of whether two different DNA samples come from the same person. A DNA sequence represents the genetic code contained within an organism. Also, [9] states that: Genetic code is a set of sequences which define what proteins to build within the organism. The problem of string matching is that there are two strings, one is the text T [1,..., n] that is, the main string given and the other is the pattern P [1,…, m] that is, the given string to be matched with the given main string given m<=n. String matching is invariably used in real word applications like Database schema, Network systems. There are 2 main techniques of string matching. The first technique is exact matching such as Needleman Wunsch, Smith Waterman, Knuth Morris Pratt (KMP), Dynamic Programming, Boyer Moore (BM) and the other is approximate matching such as Fuzzy String Searching, Karp Rabin, and Brute Force [10].

2.3 Overview of Sequence Alignment Algorithms

2.3.1 Boyer Moore Algorithm (BM)

Pandiselvam, Marimuthu, and Lawrance state that: the BM string search algorithm is an efficient string searching algorithm that is the standard benchmark for practical string search literature. The BM algorithm is considered as the most efficient string-matching algorithm in usual applications, for example, in text editors and commands substitutions. The reason is that it woks the fastest when the alphabet is moderately sized and the pattern is relatively long [11]. Singla and Garg were exploiting different kinds of sequence alignment algorithms for strings. They concluded that the Boyer Moore matching algorithm is the best for string matching [10]. Also, Jain and Pandey analyzed various kinds of multiple sequence alignment pattern matching algorithms based on different parameters such as space, time and order to find the match and their accuracy. They introduced a classical pattern matching algorithms and tools are developed from them

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

for commercial purposes. They compared the different pattern matching algorithms with their characteristics such as total number of comparisons and shift factor. They concluded that Boyer Moore algorithm is more efficient one to apply on heterogeneous system for pattern matching [12].

2.3.2 Needleman Wunsch Algorithm

Needleman Wunsch Algorithm achieves a global alignment on two sequences and it is commonly used in bioinformatics to align protein or nucleotide sequences [11]. The NW algorithm is an example of Dynamic programming, and was the first application of dynamic programming to biological sequence comparison [10] [13].

2.3.3 Smith Waterman Algorithm

It is a well-known algorithm for performing local sequence alignment; that is, for determining similar regions between two nucleotide or protein sequences [13]. Instead of looking at the total sequence, the Smith Waterman algorithm compares segments of all possible lengths and optimizes similarity measure. Figure 1 shows an example of Pairwise Sequence Alignment where two sequences are aligned and its characters are compared for a match or mismatch.

Figure 2 shows the difference between Global and Local Sequence Alignment [3]. The Global Sequence Alignment matches each character in the sequences with their positions and the length of the sequences are equal. The Local Sequence Alignment matches region of common characters in the sequences and the length of the sequences can be varied.

The paper [14] said Teiresias algorithm is an efficient and scalable algorithm that was invented by the IBM Computational Biology Group at IBM TJ Watson Research Center and first published in 1998. The Teiresias algorithm is a combinatorial pattern discovery algorithm which can be used to search for variable length patterns. Such patterns may take the form of a real character or residue, followed by some combination of dots (any character) and real characters, followed by a final residue. The ability of the algorithm for not generating simple strings like other tools is one of the features. That is, the generated patterns can have variable width, as well as may contain ambiguous characters, evaluating to motifs and not just simple words. Also, [14] said Teiresias is applicable in pattern discovery in a variety of areas, both biological, and non-biological. The Non-biological application area of Teiserias algorithm published by Nau, was used as the basis of a masquerade detection system used for network security. Specifically, the algorithm is used to initialize a gene library for computer viruses to improve the intrusion detection system.

Erbacher, Prakash. Couraud and Clash briefly explained IPAM (Incremental Probabilistic Action Modeling) to be another method that was used by Davidson and Hirsch to build an adaptive command line interface on, is also based on one-step command transition probabilities estimated from the training data. In this approach, a compression method was tested based on the premise that test data appended to historical training data compress more readily when the test data stems indeed from the same user rather than from a masquerader [15].

Herve Debar observed "Neural networks are algorithmic techniques used to first learn the relationship between the two sets of information, and then "generalize" to obtain new input-output pairs in a reasonable way". Neural networks could be applied in knowledge-based masquerade identification systems to detect the attacks and seek them in the audit stream. Herv´e Debar said "neural networks have mainly been used to learn the behaviour of actors in the system (for example, users) in intrusion detection". Neural network has an edge over statistics because of its simple approach in expressing nonlinear associations among variables and the automatic classification of the neural network. Experiments have been carried out using a neural network approach to predict the behaviour of users [16].

The researchers in [8] mentioned another approach known as Naïve Bayes. It is a one step markov method. They said the method is dependent upon the "single step transition" from one command to proceeding command and provides two conversion matrices for every user that is in upbringing data set and examining data set. When the difference between these two matrices are noticed the alarm triggers. But this approach did not provide satisfactory false alarm rate. The drawback is that this approach does not update the user profile accordingly.

It was highlighted that Hidden Markov Model (HMM) was used to compute and formulate the effectiveness of masquerade detection and it was also used to provide an efficient method for intrusion detection. The Schonlau et al Dataset, abbreviated to SEA dataset was used to classify the attackers and authorized users. The SEA

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

dataset is one of the notable data source for testing the effectiveness of masquerade detection systems or techniques. Their approach seems effective than some of the earlier approaches that implores hidden markov model analysis [17].

Cross-Semi-Global algorithm shows how gap penalties are to be determined in the command blocks to get individual user behaviour. The Cross-Semi-Global algorithm works on the principle of selecting the highest of all scores and determines the gap scores that produce the highest score. The highest score then represents the optimal score for that alignment. During the training of the masquerade detection system, Cross - Semi - Global algorithm is applied to analyze and determine the suitable scoring system for a particular user. The Cross - Semi - Global algorithm first makes several calls to the ordinary semi-global alignment while varying the values of i and j to compute the score for each combination of (i, j). It then proceeds further to compute the (i, j) combination that produces the maximum score, the first occurrence of such (i, j) combination is returned as the Peak Point. This then makes the value of i and j as penalties for inclusion of gaps in the user block and the test block respectively [18].

Changes were made in semi global positioning to avoid the limitation of false positive rate that is observed in the previous approaches in masquerade detection system. Two marking systems were introduced such as "command grouping" and "binary marking system". These marking systems are used to fix the positioning score and the gap insertion penalties [19]. The binary marking scheme compares the current signature series with new behaviours and user profile of the current user. They also affirmed that, this scheme provides a threshold value for every user to make sure that the signature series and the lexicon of the valid user remains protected from the masquerade attacks. This threshold score is evaluated through the snapshots of the user signature.

Data Driven Semi Global Alignment (DDSGA) which relied upon the Enhanced-Semi-Global Alignment was proposed. "The main strategy of DDSGA is to align the user active session sequence to previous one of the same user and label the misaligned areas as anomalous" [20]. DDSGA tries to avoid small mutation in user command.

2.4 Semi-Global Alignment Algorithm

In the works presented in [18] and [19] on Semi-Global Alignment Algorithm, the signature sequence, which represents the user's typical command behaviour, is known to be the Usersignature. The monitored command sequence, which may contain a possible substring of intruder commands, is referred to as the IntrBlock (tested block).

To use string matching in the detection of a masquerading user, n improvement on Smith Waterman local matching algorithm adapted to formulate a Semi-Global Alignment. A purely local alignment is ignored to characterize similarity between command sequences because both a prefix and suffix can be omitted in both sequences.

For masquerade detection, it is essential that all basic of the test signature of arguments are matched with the user's signature. If large prefix of the test signature and large suffix of the tested signature of arguments were to be made insignificant then the masquerade itself would be ignored. The limitation with the use of an uncontaminated global alignment is the possibility of wide segments of the signature that do not match with a portion of the user's commands. Thus, a scoring system is needed to be developed that will reward the alignment of characters in the user segment which will not penalize the mismatch of large segments of the signature. In the remainder of this section the signature sequence, which represents the user's typical command behaviour, is known to be the Usersignature. The monitored command sequence, which may contain some substring of masquerader commands, will be referred to as the IntrBlock (tested block).

2.5 Cross-Semi-Global Alignment Algorithm

Improvement was made on the semi-global alignment algorithm by introducing an algorithm to obtain optimal values for user gap and intrusion gap to increase the hit rate [18]. Figure 3 shows the Semi-Global Alignment Algorithm and figure 4 shows the Cross-Semi-global Alignment Algorithm.

III. METHODOLOGY

The Cross-Semi-Global Alignment Algorithm was improved upon to give optimal values for User Gap score and Test Gap score. The Semi-Global Alignment and the Hybridized-Cross-Semi-Global Alignment Algorithms were evaluated. Java programming language was used to implement the algorithm and for the system evaluation. The SEA dataset [21] [22] was downloaded and the system's performance was evaluated.

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

3.1 Process Model

Evolutionary Process Model was used for the descriptive and diagrammatical representation of the system life cycle. The model was used because researchers get a chance to experiment partially developed system. Also, it reduces the error because the core modules get tested thoroughly. Figure 5 shows the Research Methodology Process.

3.2 Intrusion Detection Concept

This is the process of collecting data and information about users and building a profile for each user. After the completion of the process, the system matches the legitimate user sequence of commands and the set of commands that come from masquerader and cross validate it with logs in the profile. The behaviour that does not match that of the profile is flagged an attack. For a masquerader to access resources on the victim's computer, he would have got the login credentials of a legitimate user unlawfully. Also, the intruder would have equal rights and privileges and would be bound by security policies enforced on the entire system. The user signatures were monitored over a period of time. Within the period, the behaviours of user have been collected with slight differences with the use of commands on the system. The masquerader will attempt to mimic the behaviour of the legitimate user in order to access the data on the computer system for malicious activities without the consent of the user [3]. At the point of navigating through the system, the system will flag Intrusion alarm to the administrator to alert him.

3.3 Architectural Design for the Intrusion Detection System

The user input access credentials to the computer system regardless of the location of the computer system. If the login credential is valid at the Access Control Engine, the identity of the user will be verified at the Users' Profile repository. If the behaviour of the user corresponds to the profile, he or she will be granted access to perform operation. Otherwise, he or she will be directed to the Administrator for verification, if found a legitimate user, the user's signatures will be updated in the Users' Profile. Also, if the login credentials of the user is not valid, the user will be directed to the Administrator for access credential update for genuine user, else, the System will trigger Intrusion Alarm as shown in Figure 6.

In Figure 5, the third stage (Design of Architecture and System Core) and Figure 6 (Data Repository) describe the working process of existing system. The existing system implemented Semi-Global Alignment and Cross-Semi-Global Alignment Algorithms. The system detects an attacker at the point of manipulate files on the system of a legitimate user.

3.4 Parameters

The improvement that was made on the Cross-Semi-Global Alignment Algorithm was the tuning of the algorithm as it was represented as "Tune the Parameters" process stage in Figure 5. See Figure 7 for the improvement on the Cross-Semi Global Alignment Algorithm which results to Hybridized-Cross-Semi-Global Alignment Algorithm. This new algorithm was combined with traditional Semi-Global Alignment Algorithm. The variables that were used to execute the matching process in the Intrusion Detection System are six; User Block, Test Block, Match Score, Mismatch Score, User Gap Score and Intrusion Gap Score. The first four scores are generally referred to as the Scoring System Parameters while the last two are manipulated scoring system. This work made a significant improvement on the Cross-Semi-Global Alignment Algorithm to get optimal User Gap Score and Intrusion Gap Score.

User Block/Signature: is the distinct set of commands or characters of legitimate users that was collected from SEA dataset.
Test Block/Signature: is the polluted set of commands that is classified as masquerade in the SEA dataset.
The Match Score: is a positive score for reward of perfect match of commands in the test block and the user signature.
The Mismatch Score: is a penalty for any mismatch in the command sequence of the test block and the user signature.
The Intrusion Gap Score: is the penalty for presence of gaps within the test block
The User Gap Score: is the penalty for presence of gaps within the user block

The systems use the combined aforementioned four parameters (the User Block, the Test Block, the Match Score, the Mismatch Score) by alignment algorithm to compute the optimal score. The optimal score is a value function from the alignment of the test block and the user sequence. This optimal score that was received from the alignment was then compared with the threshold score to determine if the test block is an intrusion block or not.

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

3.5 Threshold Sequence and Score

The threshold sequence score was used as the benchmark score for each user. The score was generated from the learning phase and stored inside the User's Profiles labeled in Figure 6. It provides a window for users to use unique or uncommon commands and provides update. The range varies between 10% – 30%. In this work, the scores of this alignment are computed using the scoring values in table 2.

Nevertheless, the Intrusion Gap and Signature Gap are left to Hybridized-Cross-Semi-Global Alignment to determine from the training of the intrusion detection system on the normal user threshold sequence of commands.

3.6 Experiment Metrics and Test

This test focused on the effects of changing the various parameters of the alignment algorithm on the false positive and false negative rates. This approach makes a significant contribution to intrusion detection because of it manipulating parameters. To justify this approach with other masquerade detection algorithms, metrics like False Positive Rate, False Negative Rate (also known as True positive), and the Hit Rate were adapted to check the efficiency of the Hybridized-Cross-Semi Global Alignment Algorithm over other alignment methods.

i. A False Positive: is a block that does not include any attack but the algorithm marked it as having a masquerade.

ii. A False Negative: is a block that has a masquerade but the algorithm marked it as not having a masquerade.

iii. A Hit is an attack block and the system properly marked it as having a masquerade.

The statements below summarize the metric calculations used by the algorithm.

Metric 1: False Positive Metric [3]**:**

$$\text{False Positive}_{\text{overall}} = ([\sum_{i=0}^{n}(fi/ni)]/\text{u}) * 100 \quad \ldots \text{ equation 1}$$

where fi denotes number of false positives;
ni denotes number of non-intrusion command sequence blocks, and
u denotes number of users (50 in this case).
n denotes the total number of command sequence blocks

Metric 2: False Negative Metric [3]**:**

$$\text{False Negative}_{\text{overall}} = ([\sum_{i=0}^{n}(fni/ni)]/\text{u}) * 100 \quad \ldots. \text{ equation 2}$$

where fn denotes number of false negatives;
ni denotes number of non-intrusion command sequence blocks, and
u denotes number of users (50 in this case).
n denotes the total number of command sequence blocks

Metric 3: Hit Rate Metric [3]**:**

$$\text{HitRate}_{\text{overall}} = 100 - \text{FalseNegative}_{\text{overall}}. \quad \text{..equation 3}$$

False Positives, True positive and Hits were computed for each user in the profile base, and translated into proportional rates, and the sum and average values for all iteration are evaluated for the 50 users.

3.7 Command Mismatch Scoring

The semi-global alignment algorithm was used and performs to some extent without rewarding or penalizing mismatches. These mismatches can be leveraged on to better check how the extent at which the masquerade block matches with the user's profile, and as a result, channel our technique to the limitation of masquerade

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

detection. For customized mismatch scoring system, Hybridized-Cross-Semi-Global Alignment Algorithm was used to allow for the possibility that the legitimate user may have interchanged one command with another in a particular alignment. This allowed the system to punish commands that are not as likely to be interchanged while rewarding commands that have a good probability to be interchanged with each other.

Mismatch Score Metric [3]:

M = Mismatch score
S = # of occurrences of the intrusion block command in the user's signature
C = # of distinct commands in the user's signature
M = [S / (5000/C)] - 1
If (M > = 1) { M = 1 }

3.8 Legitimate User and Masquerader/Intruder Behaviour Patterns

An intrusion detection system using pattern matching of user behaviours has to be trained with the sequence of audit generated from the user behavioural pattern of path towards the information system. This process of training the masquerade detection system (or IDS) is described as the Learning Phase of the system. Once the system has learnt the behavioural pattern of the legitimate user, the score could be provided (often known to be threshold score) for his behaviour based on the audit data. Subsequent behaviours of the user around the system will then be audited and scored. The best score (optimal score) will be checked with the user threshold score. It is assumed that, the best score of the normal user activities will be within the neighbourhood of the threshold score while the masquerader's score will show a significant drop from the normal user threshold score.

3.9 SEA Dataset

To facilitate the result of this study in comparism with other approach, Schonlau et.al. dataset was used. The learning phase was carried out by Schonlau et. al., and it leads to acceptable standard data for performance evaluation of algorithms in Intrusion Detection. For this study, 50 users were used to experiment the Hybridized-Cross-Semi-Global Algorithm [3] [21] [22]. Each of the 50 users was given access to the Windows Operating System based on their privileges and commands were inputted on different occasions. The system audited the logs of operations of the user and the commands were converted to ASCII coded sequence of data. Authorized user's behaviours were imitated for all the users and the average alignments were collected.

3.10 Hybridized-Cross-Semi-Global Alignment Process

Figure 8 describes the process of training and updating legitimate user's data and validation of authenticity of commands that were used on the system.

i. **Data Set Collection:** This phase collects users signatures separately
ii. **User Sequence:** This phase collects individual signature for different sessions and break them into subsections
iii. **Test Sequence:** Alignment of substrings are carried out in this phase
iv. **Comparing of Data:** This phase compares each user's signature against intrusion
v. **Scoring Values:** This is the optimal value for each user for alignment
vi. **Attacker:** This is the sequence of masquerade that is trying to gain access to the computer network
vii. **User:** This is the legitimate user of a computer system
viii. **Update User Sequence:** This phase is done by Administrator in case user's signature changes.

3.11 Window ASCII File

The Window ASCII file corresponding to the SEA dataset is given in appendix A. The file contains 100 rows and 50 columns. Each column corresponds to one of the 50 users of SEA dataset. Each row of the commands corresponds to a group of hundred (100) commands. The commands begin from position five thousand and one (5001) and end at the position fifteen thousand (15000). The 0 and 1 are used as the entry bits in the files. The 0 signals that the corresponding hundred (100) commands are not containing any attack by a masquerader and a 1 bit signals that the commands are having intrusion.

3.12 Algorithm Implementation

This stage in the Evolutionary Model in figure 6 above is where the new algorithm was implemented. It is evaluated

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

to check for performance by stakeholders and can also be deployed in the real environment. It is an iteration process part of which this research was based upon. The Algorithm undergoes Tuning process for better performance. The results are evaluated till there is conventional agreement that it has reached the optimal limit.

Java programming language was used to implement the algorithm for proper simulation. During the implementation, the Hybridized-Cross-Semi-Global Alignment was simulated first to get the optimal user gap and intrusion gap scores since each user has different behaviour. The variables, user_gap and intrusion_ gap were passed as a parameter to Semi-Global Alignment Algorithm. This process was iterated for the 50 users. The GUI was designed to test the performance. The SEA dataset was used to simulate the system and the results showed that the algorithm performed well with hit rate of about 90% and false positive of about 4.1 %. Windows ASCII file and masquerade file were downloaded from the link below to verify the test:
http://www.schonlau.net/masquerade/masquerade_summary.txt

IV. RESULTS AND DISCUSSION

Having identified the need to improve the security of data stored on the system, there is a need to develop a system that will detect intrusion. The Hybridized-Semi-Global Alignment Algorithm was developed to improve on the security of data. This is done by increasing the hit rate and reducing the false positive.

The output for the values for User Gap and Intrusion Gap in Hybridized-Cross-Semi-Global Algorithm was passed to Semi-Global Algorithm. The implementation of the new algorithm was demonstrated using Java programming language due to its support for object oriented design, portability and user interface flexibility. Also, SEA dataset with profiles of 50 users and Windows ASCII file were used to test the performance of the new algorithm.

System GUI Design and Partition
To simulate the new system, the GUI was designed. The system was divided into three components; Server, Client and Masquerade Attacker as shown in figure 9.

Server
This component has a profile of the legitimate user signature (50 users). The legitimate clients can connect and manipulate files as they wish. The Server Interface is shown in figure 10.

Client
This component is an interface where any user be it an authorized user or an attacker can make attempt to login to the system, send messages and carry out any other activities if he is privileged. The Client Interface before user login is shown in figure 11.

The Client Interface after User Login is shown in figure 12.

Masquerade Attacker
This component is an interface that checks the percentage of hit rate to determine if the Client is a legitimate user or an intruder and the graph is drawn as shown in figure 13.

The graph shown in figure 14 plots the similarity rate of the alignment of the sequences.

Overall Performance
The performance of Hybridized Semi-Global Alignment and Cross-Semi-Global Alignment Algorithm was measured in hits and false positive. The experiment

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

yielded a hits rate of about 90% and a 4.1% of false positive. This was obtained by comparing a masquerade location in the command blocks. Table 4 shows the results of Hybridized Semi-Global Alignment and Cross-Semi-Global Alignment Algorithm.

The results shown in Table 4 reflect the efficiency of the Hybridized-Cross-Semi-Global Alignment.

The results shown in Table 5 reflect the efficiency of the Hybridized Semi-Global Sequence Alignment Algorithm when compared with other alignment algorithms.

In Table 5, the performance of the Hybridized-Cross-Semi-Global Alignment Algorithm is being compared with the performance of other alignment algorithms as found in the literature [3] [18] [20].

Overall Performance
The performance of Hybridized-Cross-Semi-Global Alignment and Semi-Global Alignment Algorithm was measured in hits and false positive. The experiment yielded a hits rate of about 90% and a 4.1% of false positive. This was obtained by comparing a masquerade location in the command blocks. Table 4 shows the performance of the Hybridized-Cross-Semi-Global Alignment Algorithm. In table 5, the performance of the Hybridized-Cross-Semi-Global Alignment Algorithm is being compared with the performance of other alignment algorithms.

V. CONCLUSION

Bioinformatics is a field in computer science that can help in solving varied number of problems in many spheres of life. It possesses ability to design algorithms that can be tuned to solve a specific problem in a niche, Intrusion Detection in our case. The study has successfully presented a working implementation of a Hybridized-Cross-Semi-Global Sequence Alignment Algorithm for the detection of intruders, and shows that, with appropriate adaptation and manipulation, it performs comparably with the top masquerade detection algorithms stated earlier. While an obvious and critical amount of manipulation has been done to the generic Smith-Waterman local alignment algorithm to produce a good intrusion detection algorithm, some other metrics were used for improvement in our algorithm's performance. Ability for alignments to produce many different statistics makes it to characterize similarity between command sequences. These statistics include the density of the matching (alignment value divided by alignment length), the maximum, minimum, and average gap length, the overall figure for matching and mismatched sequences, in addition to the overall figure for gaps in each of the matched subsequences. These evaluations resolve different aspects of the alliance which can be harnessed to differentiate the user substrings from attacker substrings.

Contribution to Knowledge
Having carried out this study, there are significant inputs between the result of this study and the results from other researchers. They are:

- The increase in hit rate between the Hybridized-Cross-Semi-Global Alignment Algorithm and the Cross-Semi-Global Alignment Algorithm is 7.9%.
- The decrease in false positive rate is 1.3%.

Recommendations
This work focused mainly on improving the Semi- Global Alignment Algorithm to increase the percentage of hit rate of matched sequences and reduce the index of false positive and false negative. This system can be adopted and be deployed to add to security of computer systems.

Future Works
This study was able to improve the Semi-Global Alignment Algorithm, but researchers can improve on this by combining Semi–Global Alignment and other Techniques for possibility of having higher percentage of hit rate and to also reduce the processing time.

ACKNOWLEDGEMENT

We are using this medium to acknowledge and thank the Reviewers for their thorough and in-depth revision of the paper.

REFERENCES

[1] Jaeger T. (2009). Gasserbook, 1-12. Retrieved from https://doi.org/10.2200/S00126ED1V01Y200808SPT001

[2] Stallings W. (2007) Cryptography and Network Security (4th Ed). Available at http://www.inf.ufsc.br/~bosco.sobral/ensino/ine5680/material-cripto-seg/2014-1/Stallings/Stallings_Cryptography_and_Network_Security.pdf

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

[3] Coull S., Branch J., Szymanski B. and Breimer E. (2003), Intrusion Detection: A Bioinformatics Approach. Proceedings of 19th Annual Computer Security Applications Conference, Las Vegas, NV. Pp 24-33.

[4] Salem M. B and Stolfo J. S. (2011). Modeling User Search Behaviour for Masquerade Detection. Retrieved from, 1-20.

[5] Denning D. E. and Denning P. J (1997). Cyberspace attacks and Counter measures. *In Internet Besieged D. E. ACM Press, New York. 29–55.*

[6] Salem M. B and Stolfo J. S (2011). Combining a Baiting and a User Search Profiling Techniques for Masquerade Detection.

[7] Youssef A. and Emam A. (2011). Network Intrusion Detection Using Data Mining and Network Behaviour Analysis. *International Journal of Educational Advancement EISSN: 1744-654, 3(6), 1-12.*

[8] Nair C. S, Lavanya S., Chaithra R, Siddiqua E. W (2016). Masquerade Detection Using Improved Semi Global Positioning Algorithm. *International Journal of Research in Engineering and Technology eISSN: 2319-1163, 5(4), 1-5.*

[9] Izzat A. and Nuser M. (2012) *International Journal of Advanced Science and Technology, 47, 1-20.*

[10] Singla N. and Garg D. (2012). String Matching Algorithms and their Applicability in Various Applications. *International Journal of Soft Computing and Engineering, ISSN: 6(1), 2231-2307.*

[11] Pandiselvam P., Marimuthu T., and Lawrance. R. A (2013). Comparative Study on String Matching Algorithms of Biological Sequences, 1-5.

[12] Jain P. and Pandey S. (2012), Comparative Study on Text Pattern Matching for Heterogeneous System, *International Journal of Computer Science and Engineering Technology, ISSN:3(11), 2229- 3345.*

[13] Gomaa N. H. and Fahmy A. A. (2012), Short Answer Grading using String Similarity and Corpus-Based Similarity, *International Journal of Advanced Computer Science and Applications,3(11).*

[14] Nau L. J. (2011). A Scalable, Memory efficient Multicore TEIRESIAS Implementation, 1-62.

[15] Erbacher R. F., Prakash S., Couraud J. and Clash C. L. (2007). Intrusion Detection: Detecting Masquerade Attacks Using UNIX Command Lines. *In Proceedings of the 6th Annual Security Conference, Las Vegas, NV, 1-18.Available at www.security-conference.org.*

[16] Debar H. (2008). An Introduction to Intrusion-Detection Systems. *IBM Research, Zurich Research Laboratory, S¨aumerstrasse 4, CH–8803 R¨uschlikon, Switzerland. Available at, 1-18.*

[17] Bhukya W. N. and Suresh K. G., (1999) "A study of effectiveness in masquerade detection," *EIGAR 1999 Best Paper Proceedings, 34-50.*

[18] Sodiya A. S., Folorunso O., Onashoga S. A, and Ogunderu O. P. (2011). An Improved Semi-Global Alignment Algorithm for Masquerade Detection. *International Journal of Network Security, 13(1), 31–40.*

[19] Coulla S. E. and Szymanski B. K. (2008), "Sequence alignment for masquerade detection," *J. Comput. Statist. Data Analysis.*

[20] Devkate A. C., Pagare V. G, More D. M, Misal S. R, Patil Y. (2015). Masquerade Detection Using Data Driven Semi-Global Alignment Approach. *International Journal of Advanced Research in Education & Technology (IJARET),* ISSN: 2394-2975, 2(4), 1-5.

[21] Schonlau Dataset, http://www.schonlau.net

[22] Schonlau M., DuMouchel W., Wen-Hua J., Alan F. K, Theus M. and Vard Y (2001). Computer Intrusion: Detecting Masquerades, 16(1), 1-17.

http://www.schonlau.net/masquerade/masquerade_summary.txt

http://www.ebi.ac.uk/Tools/psa/emboss_water/nucleotide.html.

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

Table 1: Difference between Misuse and Anomaly detection [7].

	Misuse Detection	**Anomaly Detection**
Characteristics	use patterns of well-known attacks (signatures) to identify intrusions, any match with signatures is reported as a possible attack	use deviation from normal usage patterns to identify intrusions, any significant deviations from the expected behaviour are reported as possible attacks
Drawbacks	- False negatives - Unable to detect new attacks - Need signatures update - Known attacks has to be hand-coded - Overwhelming security Analysts	- False positives. - Selecting the right set of system features to be measured is ad hoc and based on experience - Has to study sequential interrelation between transactions - Overwhelming security Analysts

```
__GTGACATGCGAT__AAGAGG___CCTT__AGATCCGGATCTT
  |  |||  || ||||  | ||       ||||   |||| | |    |
GGGAGAC_TGCGATACAAG___TTACCTTGTAGATCTG_TCTT _
Key:_ gap
    | match
```

Figure 1: Pairwise Sequence Alignment [3].

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

Global Alignment:

```
--AGATCCGGATGGT--GTGACATGCGAT--AAG--AGGCGTT
    |||  |   |    | |||||  ||||||   |||   |    | ||
GTCCATCTG--TCTTGGGTGAC-TGCGATACAAGTTA--CCTT
```

Local Alignment:

```
_ _AGATCCGGATGGT__ GTGACATGCGATA _ _AG_ _AGGCGTT
                   |||||  |||||||
GTCCATCTG_ _ TCTTGG GTGAC_TGCGATA _CAAGTTA_ _CCTT
```

Figure 2: Difference between Global and Local Sequence Alignment [3].

```
Input: string Usersignature of length m, string IntrBlock of length n
1. Initialize a matrix, D, of type integer
2.      for i=0 to m
3.          for j=0 to n
4.              if(j=0 or i=0)
5.                  D[i][j]=0;
6.              else
7.                  If (j=n or i=m)
8.                  top=D[i][j-1];
9.                  left=D[i-1][j];
10.             else
11.                 Top =D[i][j-1] – gUsersignature;
12.                 Left =D[i-1][j] – gIntrBlock;
13.                 if (top<0) top=D[i][j-1];
14.                 if (left<0) left=D[i-1][j];
15.         diagonal =D[i-1][j-1] + matchScore(Usersignaturei-1,IntrBlockj-1);
16.             D[i][j]=maximum(top,left,diagonal);
17. return D[m][n];
```

Figure 3: Semi-Global Alignment Algorithm [3].

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

```
1: Begin
3: Usersignature: normal user signature.
4: testSig: monitored command sequence for intrusion.
5: matchScore: the score for matches during the alignment.
6: misMatchScore: the score for mis-matches during the
alignment.
7: i: the score for the userGap.
8: j: the score for the intrusionGap.
9: OptimalValue(i, j): two dimensional array for storing
alignment scores corresponding to (i, j) combinations.
10:
11: // the algorithm logic is described below:
12: for i = −n to -1 step 1 do
13: for j = −n to -1 step 1 do
14: OptimalValue(i, j)=SemiglobalAlignment(Usersignature, testSig, matchScore, misMatchScore, i, j)
15: end for
16: end for
17: // determining the peak point(i, j)
18: for i = −n to -1 step 1 do
19: for j = −n to -1 step 1 do
20: if (max(OptimalV alue(i, j)) = OptimalScore)
then
21: PeakPoint = (i, j)
22: end if
23: return PeakPoint
24: end for
25: end for
26: End
```

Figure 4: Cross-Semi-Global Alignment Algorithm [18].

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

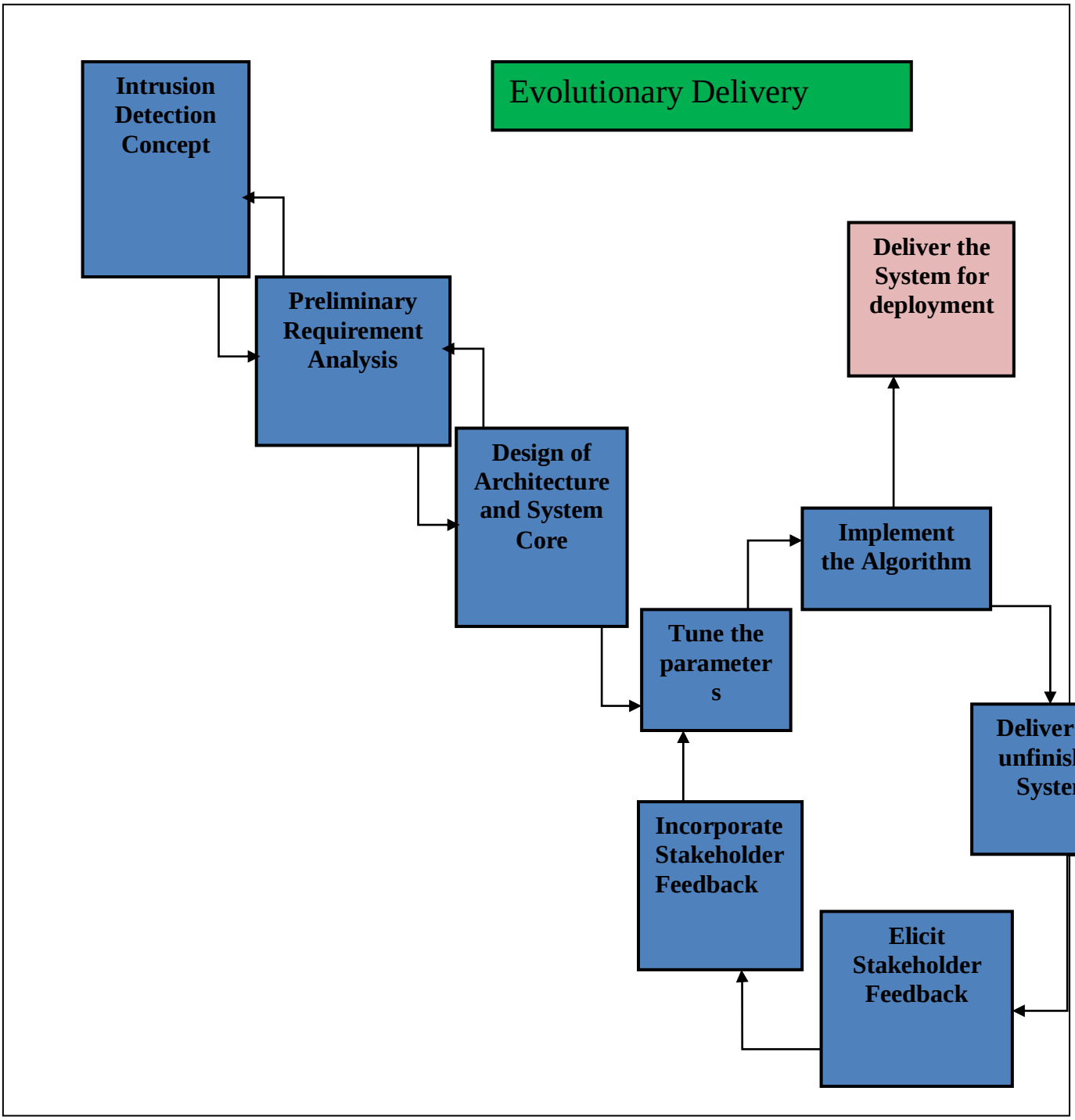


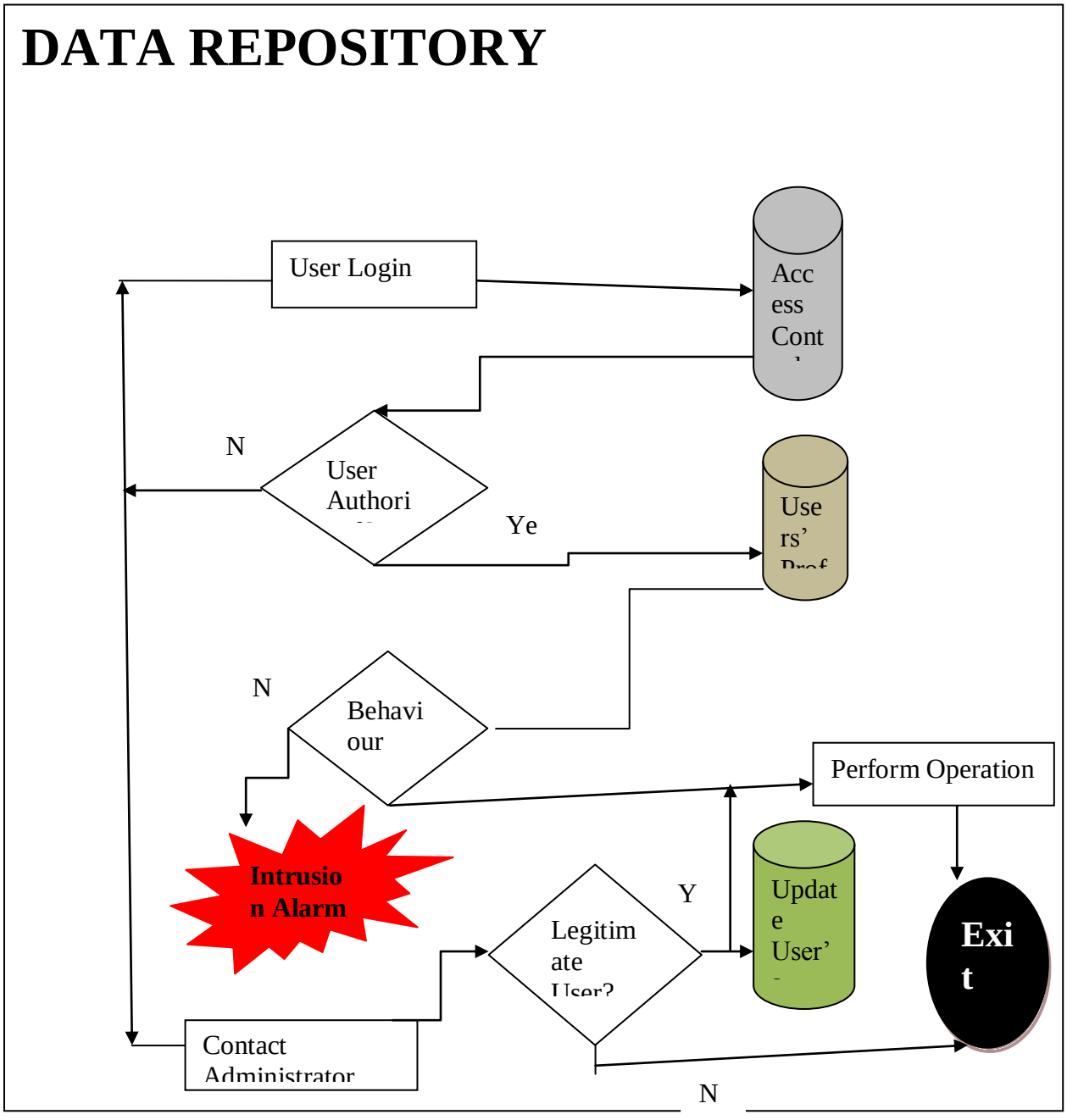
Figure 5: Research Methodology Process

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net



Connection

DATA REPOSITORY



© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

Figure 6: Architectural Design for the Intrusion Detection System [8].

Table 2: Parameters and Scores

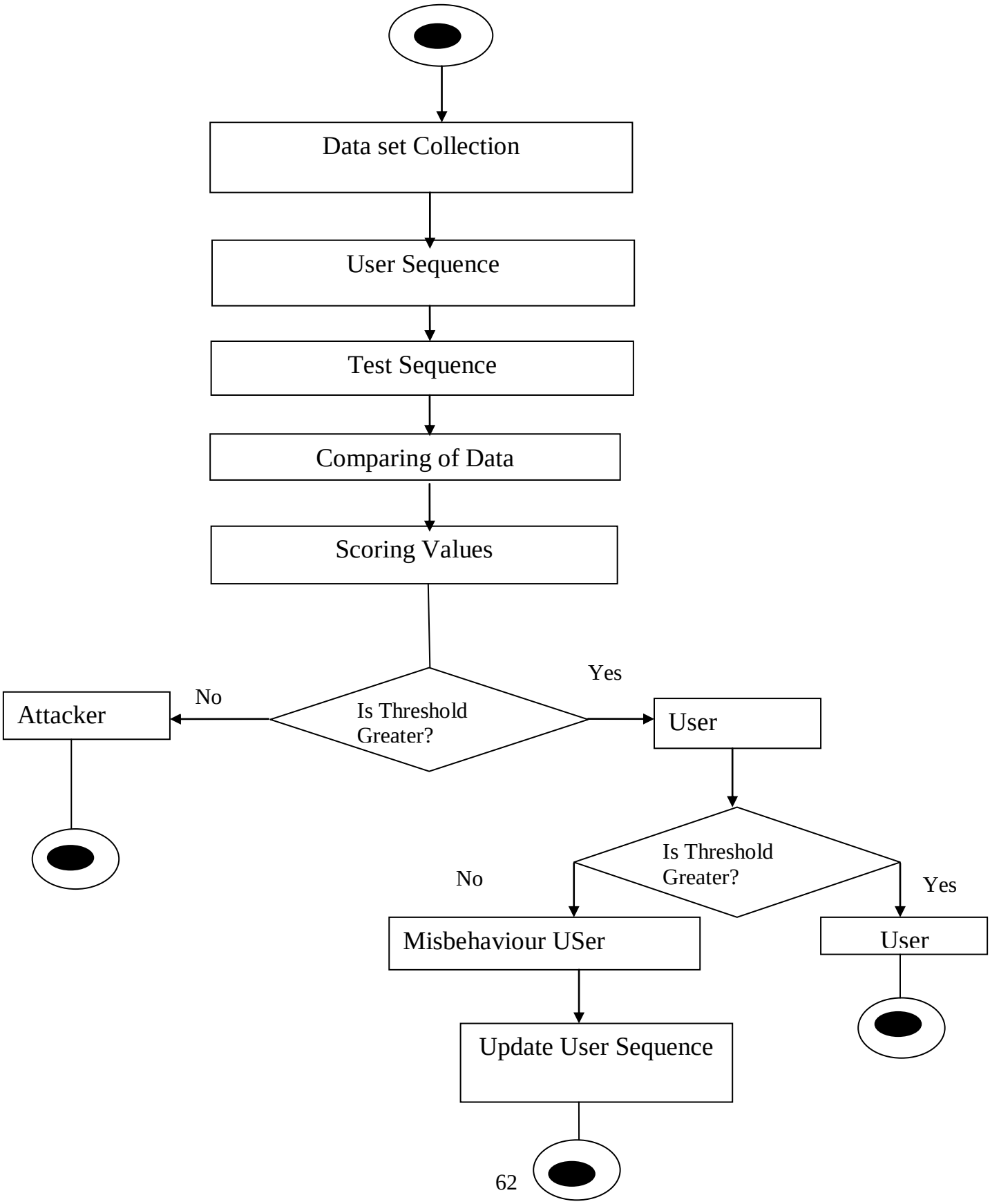
Parameter	**Score**
Match	1
Mismatch	0
Threshold	10%

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

```
1: Begin
2: // the list of method parameters are given below:
3: Usersignature: normal user signature.
4: testSig: monitored command sequence for intrusion.
5: matchScore: the score for matches during the alignment.
6: misMatchScore: the score for mis-matches during the alignment.
7: i: the score for the userGap.
8: j: the score for the intrusionGap.
9: OptimalValue(i, j): two dimensional array for storing alignment scores corresponding to (i, j) combinations.
10:
11: // the algorithm logic is described below:
12: for i = 1 to n+1 step 1 do
13: for j = 1 to n+1 step 1 do
14: OptimalValue(i, j)=SemiglobalAlignment(Usersignature, testSig,matchScore,misMatchScore, i, j)
15: end for
16: end for
17: // determining the peak point(i, j)
18: for i = 1 to n+1 step 1 do
19: for j = 1 to n+1 step 1 do
20: if (max(OptimalV alue(i, j)) = OptimalScore)
then
21: PeakPoint = (i, j)
22: end if
23: return PeakPoint
24: end for
25: end for
26: End nation of (i, j).
```

Figure 7: The Hybridized-Cross-Semi-Global Alignment Algorithm

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net



© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

Figure 8: Process of Training and Updating legitimate user's data and validation of authenticity of command.

Table 3: Difference between the Cross-Semi-Global Alignment Algorithm and Hybridized Semi-Global Sequence Alignment Algorithm

Line Code	Cross-Semi-Global Alignment Algorithm	Hybridized Semi-Global Sequence Alignment Algorithm
12	for i = −n to -1 step 1 do	for i = 1 to n+1 step 1 do
13	for j = −n to -1 step 1 do	for j = 1 to n+1 step 1 do
18	for i = −n to -1 step 1 do	for i = 1 to n+1 step 1 do
19	for ij= −n to -1 step 1 do	for j = 1 to n+1 step 1 do

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

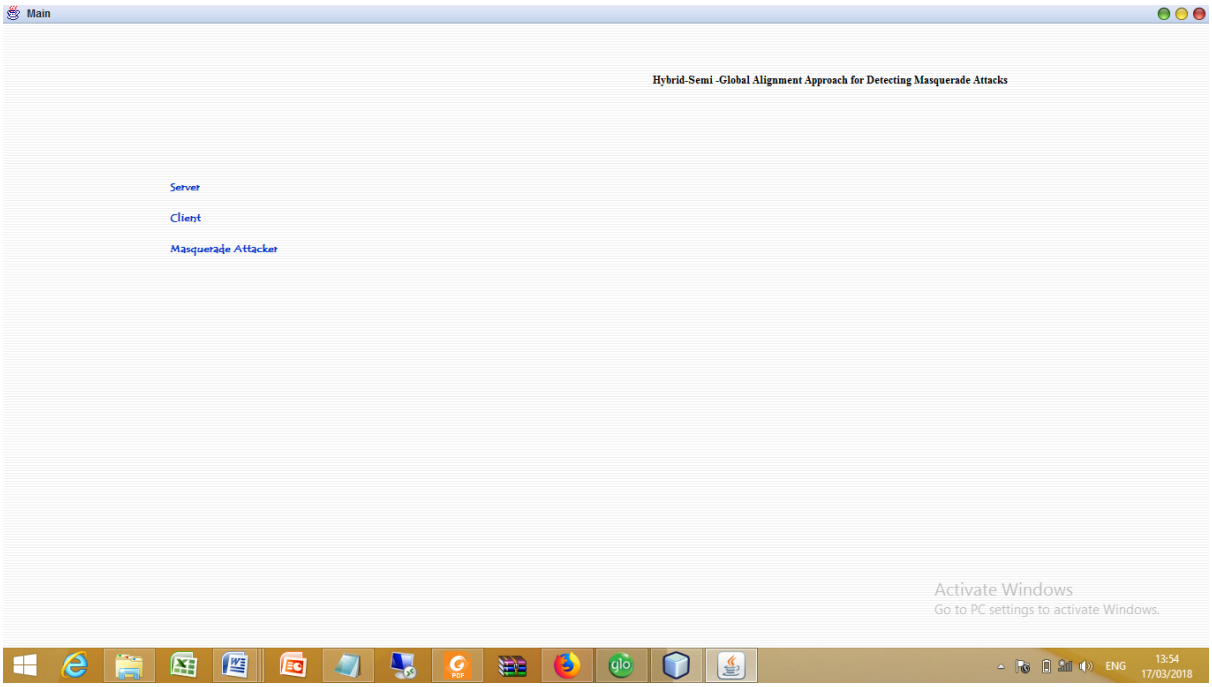


Figure 9: The System Interface

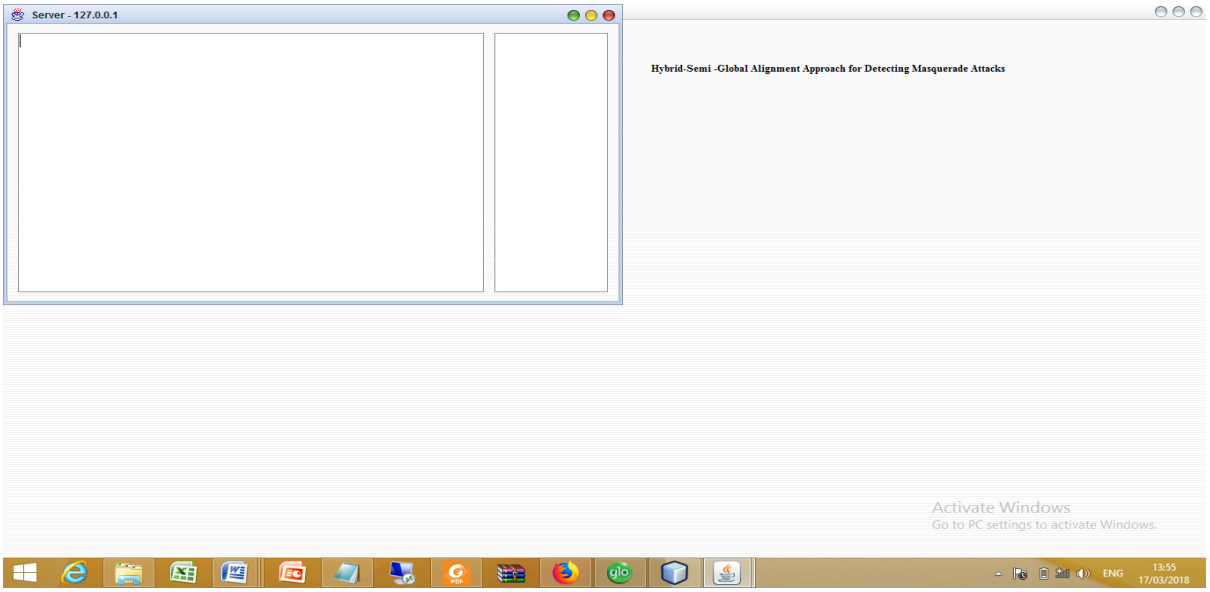


Figure 10: The Server Interface

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

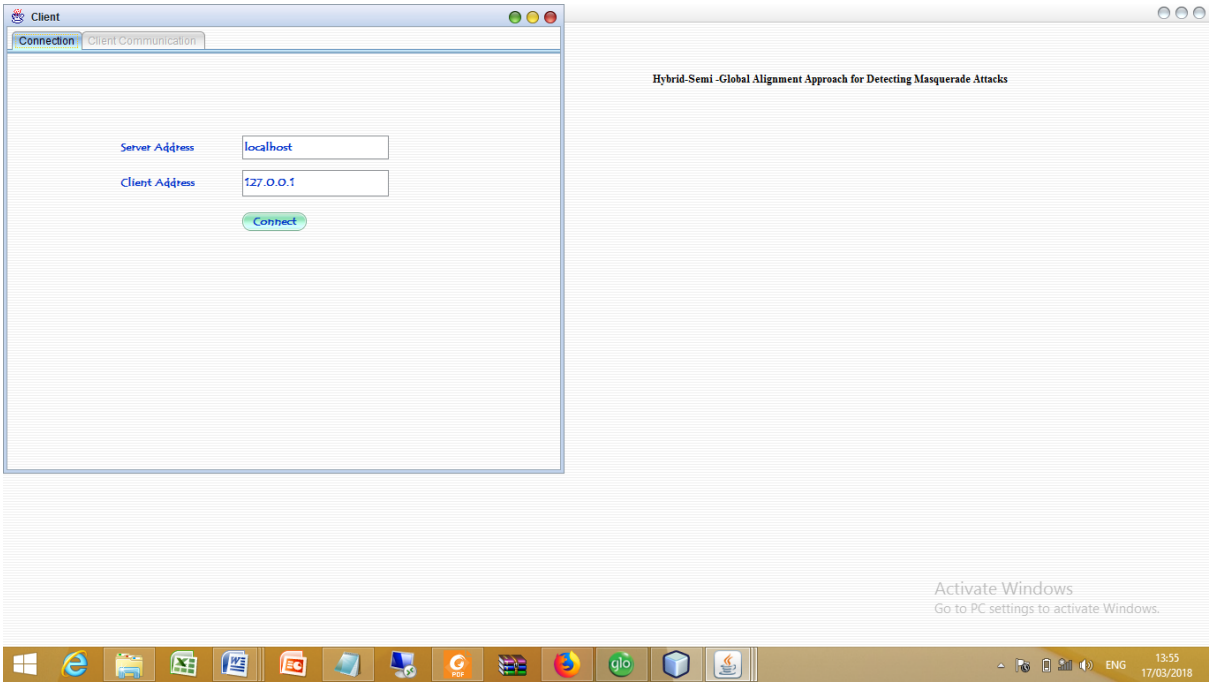


Figure 11: The Client Interface before User Login

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

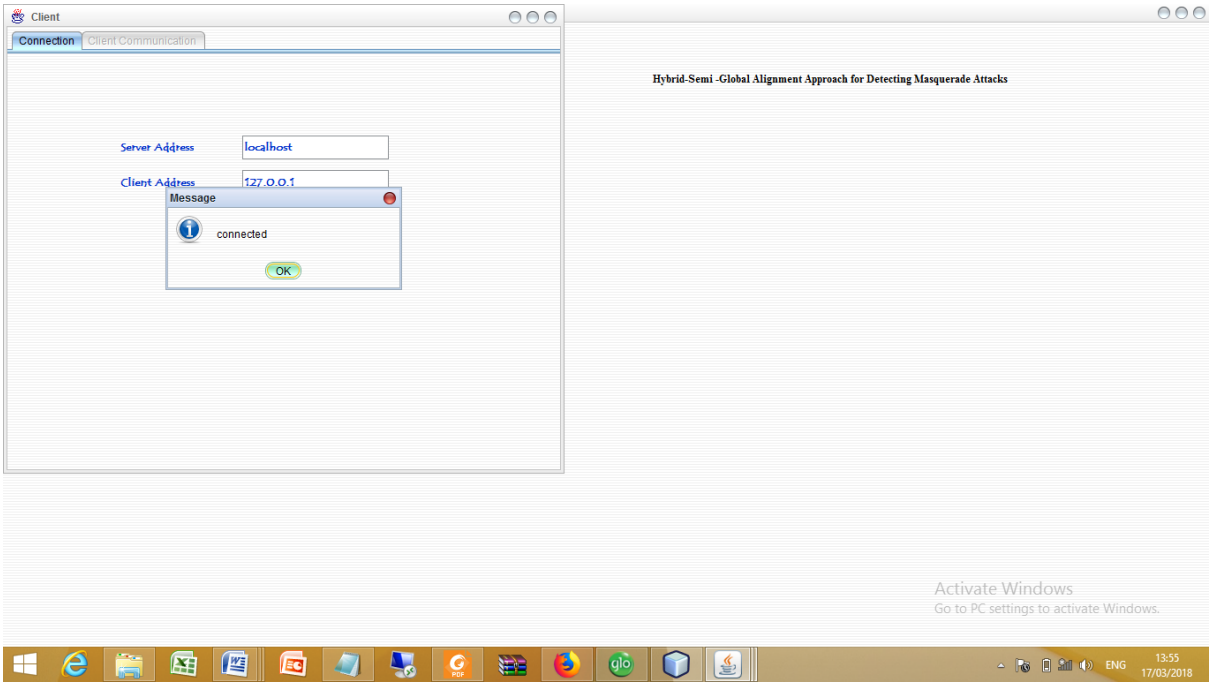


Figure 12: The Client Interface after User Login

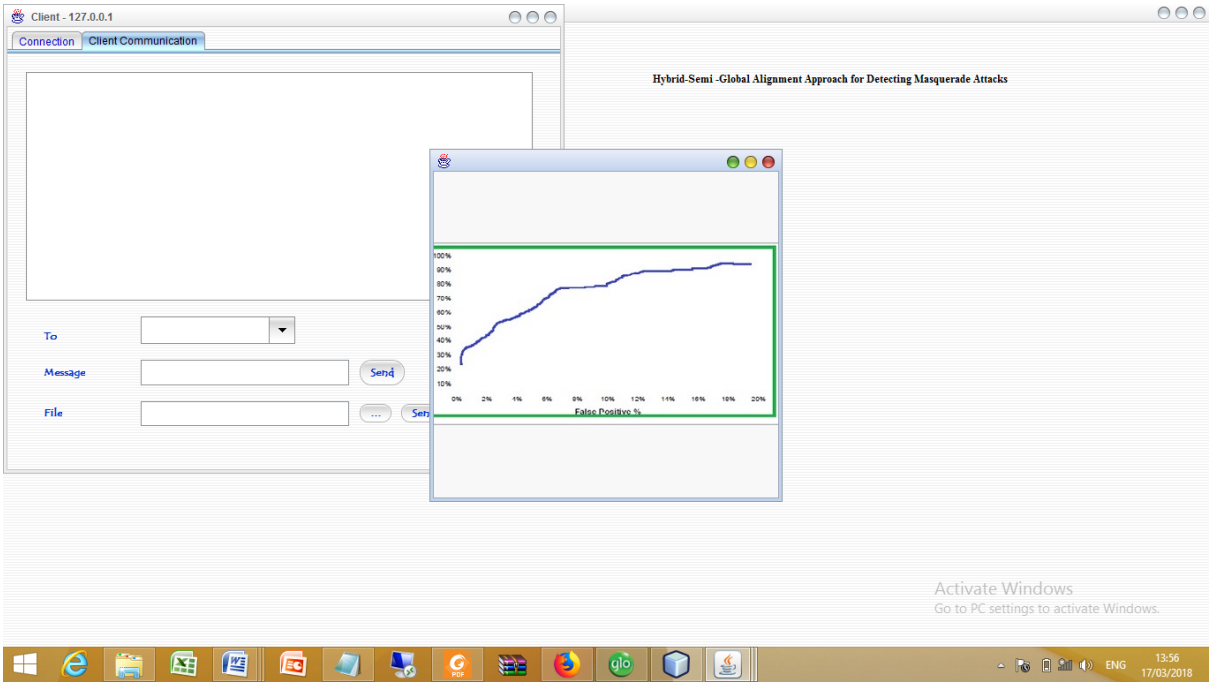


Figure 13: Checking the Hit Rate.

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

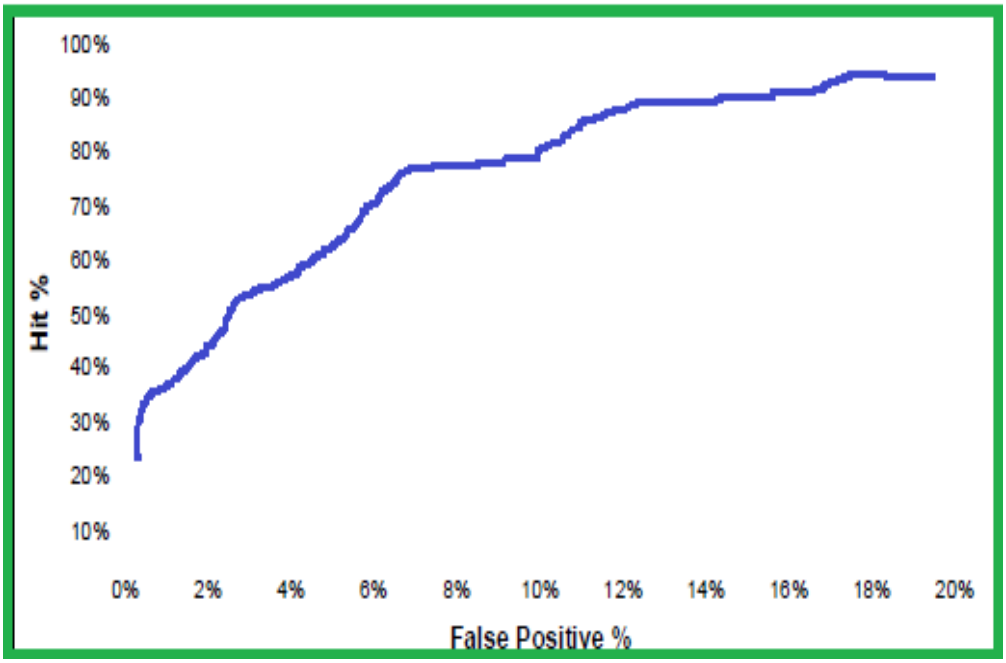


Figure 14: The Alignment Rate Graph

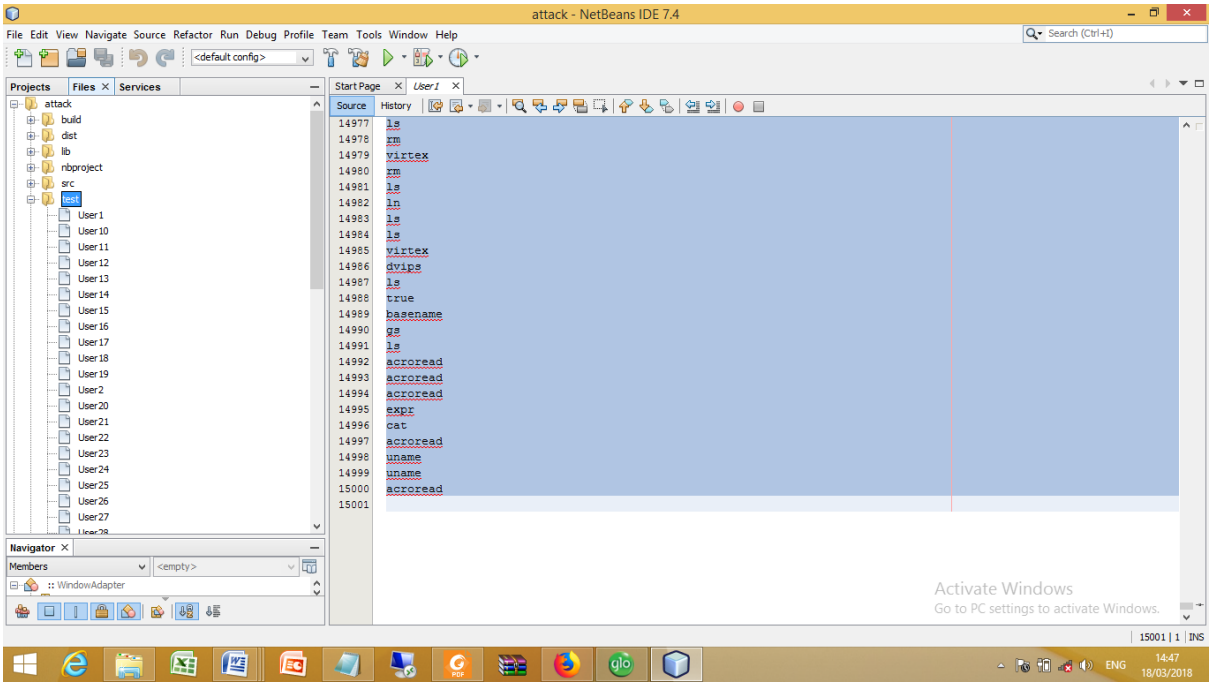


Figure 15: User1 Profile (Uncompressed Commands, Equivalent of Figure 7 above)

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

Table 4: Performance of Hybridized-Cross-Semi-Global Alignment Algorithm

Data	**Schonlau Dataset**
Type of Data	Truncated
Training Data	5000 commands
Testing Data	10000 commands
Block Size	100
Hit Rate	90%
False Positive	4.10%

Table 5: Comparison with other Alignment Algorithms

Algorithm	Hit Rate	False Positive
Hybridized-Cross-Semi-Global Alignment	90.00%	4.10%
Data – Driven Semi- Global Alignment	84.40%	5.20%
Cross Semi Global Alignment	82.10%	5.40%
Semi Global Alignment	75.80%	7.70%
Bayes 1-Step Markov	69.30%	6.70%
Naïve Bayes (no updating)	66.20%	4.60%
Naïve Bayes (updating)	61.50%	1.30%
Hybrid Markov	49.30%	3.20%
IPAM	41.10%	2.70%
Uniqueness	39.40%	1.40%
Sequence Matching	36.80%	3.70%
Compression	34.20%	5.00%

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

APPENDIX

APPENDIX A - Window ASCII File

0 0 0

© 2018 *Afr. J. Comp. & ICT* – All Rights Reserved
https://afrjcict.net

0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0