Global Journal of Computer Science and Technology: E
Network, Web & Security
Volume 17 Issue 5 Version 1.0 Year 2017
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# A Review of Technical Issues on IDS and Alerts

By Dr. Nehinbe Joshua Ojo & Onyeabor Uchechukwu Solomon

*Federal University*

*Abstract-* The fact that swindlers can trick computer and mobile systems to commit different criminal offenses have to lead to the current advancement in the domain of Intrusion Detection Systems (IDSs). While the toolkits are growing mechanisms for monitoring, analyzing, gathering and reporting activities that can endanger computer and mobile systems, however, they are frequently subjected to series of fiery debates over the years. Thus, a wide range of taxonomy has been proposed to clarify their strengths and weaknesses. Nonetheless, researchers often reticent from critical issues associated with the "used alerts" and "unused alerts" that the toolkits can generate to warn analysts. Thus, this paper presents the progression of the above mechanisms over the years; and exhaustively explains some salient issues that were faulted in the previous reviews. Finally, we suggest various ways to improve the efficacy of the toolkits and how to lessen cases of intrusions across the globe.

*Keywords: intrusion detection system; a detector; alerts; redundant alerts; workload.*

*GJCST-E Classification: H.3.7*

AREVIEWOFTECHNICALISSUESONIDSANDALERTS

*Strictly as per the compliance and regulations of:*

# A Review of Technical Issues on IDS and Alerts

Dr. Nehinbe Joshua Ojo[α] & Onyeabor Uchechukwu Solomon[σ]

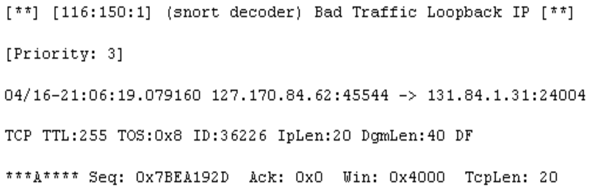
*Abstract-* The fact that swindlers can trick computer and mobile systems to commit different criminal offenses have to lead to the current advancement in the domain of Intrusion Detection Systems (IDSs). While the toolkits are growing mechanisms for monitoring, analyzing, gathering and reporting activities that can endanger computer and mobile systems, however, they are frequently subjected to series of fiery debates over the years. Thus, a wide range of taxonomy has been proposed to clarify their strengths and weaknesses. Nonetheless, researchers often reticent from critical issues associated with the "used alerts" and "unused alerts" that the toolkits can generate to warn analysts. Thus, this paper presents the progression of the above mechanisms over the years; and exhaustively explains some salient issues that were faulted in the previous reviews. Finally, we suggest various ways to improve the efficacy of the toolkits and how to lessen cases of intrusions across the globe.

*Keywords: intrusion detection system; a detector; alerts; redundant alerts; workload.*

## I. Introduction

The likelihood that companies and private individuals across the globe can lose large sum of financial and material resources to swindlers under false ploys committed with the support of mobile and computer services is of great concerns both in academia and in the industrial sector in general. These problems were envisaged in about four decades ago; and accordingly, the Intrusion Detection System (IDS) was proposed (Nehinbe, 2011). Although, the present-day Intrusion Detection Systems (IDSs) have evolved through different models, however, there are increasing concerns that new issues are constantly emerging from time to time (Ghorbani et al. 2010; Mohamed, 2013).

While various discussions and open arguments have been carried out in media and contemporary literature, some technical issues are erroneously unstressed over the years. For instance, the concept of IDS started from the work of Anderson in 1980 when the scholar classified users of mainframe computer systems into abnormal; and normal users (Anderson, 1980). Some of the existing IDSs that can be used for research purposes include Snort, Bro; and OSSEC (Stavroulakis and Stamp, 2010; Rehman, 2003; Bro, 2017).



```
[**] [116:150:1] (snort decoder) Bad Traffic Loopback IP [**]

[Priority: 3]

04/16-21:06:19.079160 127.170.84.62:45544 -> 131.84.1.31:24004

TCP TTL:255 TOS:0x8 ID:36226 IpLen:20 DgmLen:40 DF

***A**** Seq: 0x7BEA192D  Ack: 0x0  Win: 0x4000  TcpLen: 20
```

*Figure 1:* Alert from Snort on public trace file

The central issue here is that as shown in Figure 1, IDS extracts and logs attributes from every suspected packet it notices for further analysis. Unfortunately, these have also generated series of issues over the years.

An intrusion is a breach of security of a computer or mobile system (Stallings, 2011). Also, it can represent an act of unlawful access to a digital system. In this case, the location of the intruders can be inside or outside of the networks. For this reason, intruders are categorized as intruders that are insiders and intruders that are outsiders. As both names imply, the former depicts malicious users that are inside the computer or mobile networks and the latter are malicious users that are outside the computer or mobile networks.

The concept of intrusions may signify interruption of traffics in transit, stoppage or deliberate delay of services from reaching service users; invading sensitive information, destruction of components of the computer and mobile systems by causing severe damage to the software, hardware and some useful files (Kizza, 2009). Some intrusions can modify, corrupt, delete and erase directory. Accordingly, the developments of their various types often generate series of technical issues that were raised, analyzed, discussed and meticulously disputed in the past years.

The development has also lead to the evolution of standards, policies and best practices being proposed to lessen cases of intrusions over the years. In this note, qualifications, professional development and professional certifications are also emphasized as benchmarks for the recruitment of computer and mobile security professionals in some settings. Unfortunately, cases of intrusions are emerging every day. Computer users, mobile users; and community of security teams are mostly apprehensive due to the unpredictable menace of dangerous and sophisticated dimensions for compromising the security of resources reportedly occurring in some quarters globally.

Organizations and people that are victims of sophisticated intrusions can be devastated as a result of their experiences. Sophisticated intruders can swindle

*Author α σ: Federal University, Oye-Ekiti, NG.*
*e-mails: nehinbe@yahoo.com, uchechukwu.onyeabor@fuoye.edu.ng.*

people and firms funds that they have accumulated, stored and planned for the implementation or funding of projects within overnight.

Sophisticated intruders can damage corporate image and personality that have built over the years within a twinkle of eyes (Gary, 2007; Mohamed, 2013). Sophisticated intruders can intrude into the computer or mobile systems with the purpose to cheaply embarrass a wide range of community of people. They can leak sensitive information about the governments, agencies, corporate firms and highly dignified people such as celebrity and scholars to competitors, opponents; and enemies without the rethink of the consequences of their malicious behaviors on the victims.

In another dimension, there are series of overheads regarding spending, cost, apportioning of resources, control and the mechanisms necessary to promptly thwart sophisticated intrusions in a real-life environment.

Irrespective of the motives and the category of the intruders, successful and unsuccessful attacks on computer and mobile systems always leave potential dangers behind. The existence of cartel of intruders is often reaffirmed in literature. Thus, intruders may share the previous experience they have garnered with colleagues. The danger of such information sharing can be enormous if they divulge the information to dangerous and more skillful intruders that are bent to launch devastating, stealthy or destructive attacks against the previous victims.

A technical issue here is that, in the present day setting, strong IDSs will alert whenever unskilful computer and mobile users mistakenly infringe the security of other digital systems that the detectors monitor. Conversely, despite the evolutionary trend in the development of IDSs, it is improbable for the mechanism of intrusion detections to discriminate and subsequently classify attacks by the intention of each intruder.

Besides, numerous scholars have categorized IDSs into different categories. Debar et al. (2000) notably categorized IDSs by source of data, method; and concept that an IDS uses for detecting attacks. The taxonomy produced by Axelsson (2000) classified them by the detection, operations and objectives of the IDSs. In the reviewed carried out by Debar et al. (2000), misuse and anomaly detection methods are fundamental approaches for developing the IDSs. Nonetheless, as argued by Lazarevic et al. (2005) and corroborated by Scarfone and Mell (2007), IDSs lack universally acceptable classification models.

This paper exhaustively reclassifies existing IDSs on the bases of the source of data the IDS uses, the method of detection, function, structural design, the location of the detector and reporting strategies used by the IDS. Unlike the previous taxonomy, this paper explains critical and inherent issues that can maximize values and trust repose on the usage of IDSs as devices for adequately safeguarding computer and mobile systems from intrusions. Also, the paper has delved into the complexity of the intrusion detections and the existence of different methodologies for detecting malicious activities and eventually evolves better strategies for manufacturers on how they can upgrade the existing toolkits.

The remaining sections of this paper are organized as follows: Section 2 discusses the evolution of IDSs since the 1980s. Sections 3 and 4 express some of the emerging issues identified with IDS alerts and the conclusion of the paper, respectively. The latter also provides the overview of the analyses and opens up new research directions to improve the efficacies of IDSs.

## II. The Advancement in Intrusion Detection Systems (IDSs)

Debar et al. (2000), Ghorbani et al. (2010) and some scholars have proposed revised taxonomy for IDSs. However, such classifications have not explicated some technical issues recently identified while working with IDSs. Accordingly, we reclassify IDSs by the source of data that the IDS uses; the method the existing IDS use for detection of intrusions; the basic functions the IDS can perform; the structural design underpinning each IDS, the location of the detector within computer and mobile networks and various reporting strategies that the IDS used over the years. Hence, Figure 2 illustrates the schematic drawing of the proposed taxonomy to simplify the relationship between one category of IDS and another category.

### a) *Classification by source of data*

An IDS can be categorized on whether the detector obtains data from the database logs, operating system's logs, application's logs, transaction logs (in the case of financial organisations), trace files such as network traces, dump of an operating system, database and network operations and alerts from other intrusion detectors (Axelsson, 2000; Nehinbe, 2011).

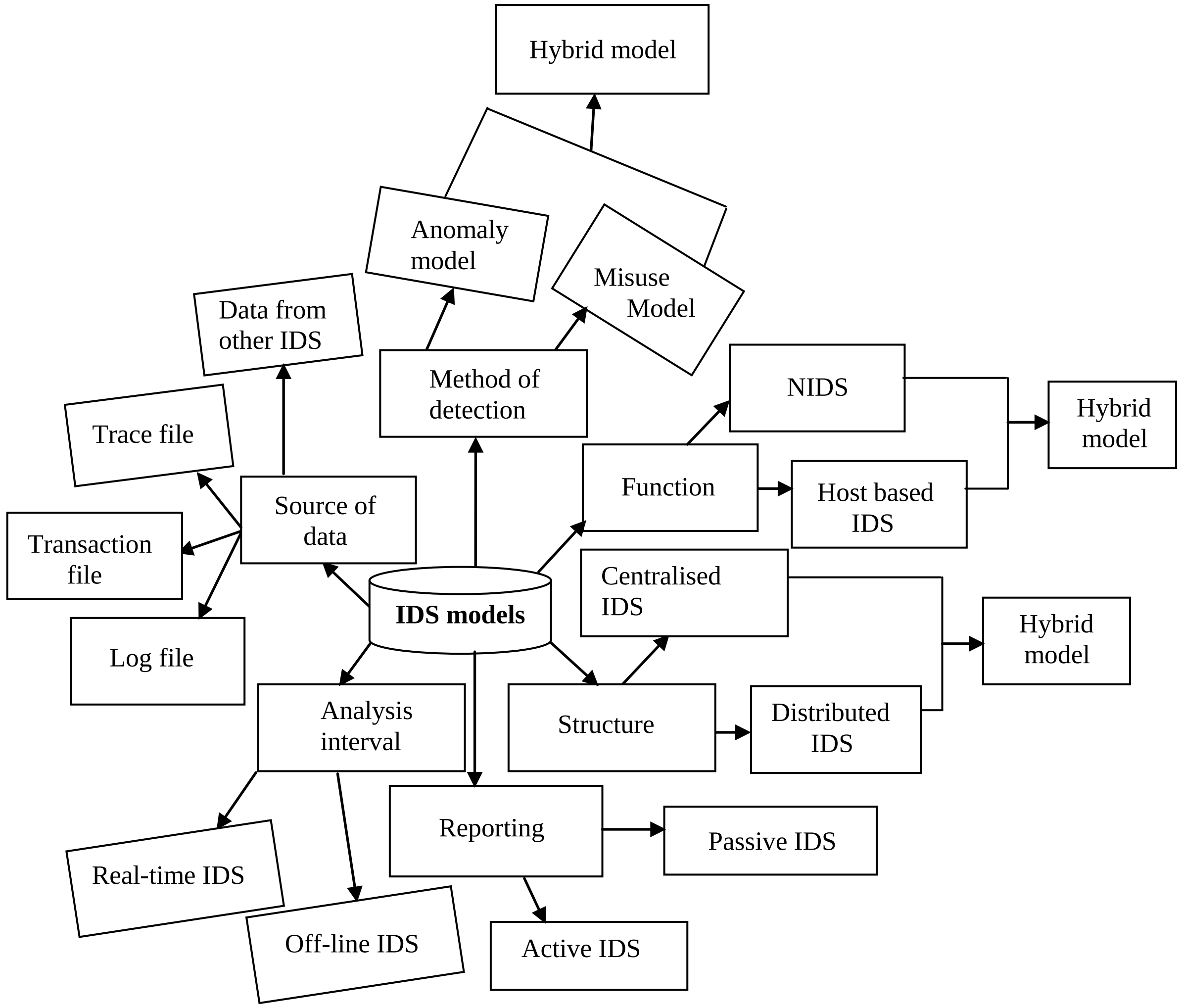


*Figure 2:* Categories of IDS

*b) Classification by function*

Different models of intrusion detectors have different capabilities. Accordingly, intrusion detectors can be categorized into host-based, network-based and hybrid intrusion detection systems (Karthikeyan and Indra, 2010). A host-based intrusion detector analyses activities of users occurring on the host computers. However, this model is ineffective to detect attacks that flood computer networks such as buffer over-flow and Distributed Denial of Service (DDoS) attacks that specialized IDS can quickly detect at the network level (Scarfone and Mell, 2007).

Contrarily, a Network-based Intrusion Detector (NID) otherwise known as Network Intrusion Detection System (NIDS) can only analyze activities of users at the network level. The detector validates each packet that migrates across its sensor with inbuilt rules or policies. Subsequently, the NIDS raises alerts to warn the presence of intrusions on the networks whenever a packet matches any of its detection rules (Amer and Hamilton, 2011). Usually, network-based intrusion detectors can also monitor activities on wired and wireless networks. Mobile network intrusion detector is a device that monitors wireless network nodes (Scarfone and Mell, 2007). However, NIDS has critical drawbacks. For instance, the strengths of NIDS depend on the capability of the rules or policies that the detector uses to detection network intrusions. Besides, the inability of some categories of the NIDS to accurately decode traffics that intruders deliberately encrypt is often a subject of contention in a realistic environment. Also, the efficacy of the NIDS to report fraudulent activities at the database, operating system and application levels is bad (Rehman, 2003).

The hybrid model integrates network-based and host-based intrusion detectors (HIDS) together. This category of detectors can concurrently monitor activities of the user both at the host level and at the network level. Nevertheless, adequate amount of capital and memory space are usually required to effectively implement HIDS in a realistic setting.

*c) Classification by method of detection*

Some intrusion detectors can detect activity that deviates from normal behavior, while others can only detect known or anticipated attacks. The former category is called anomaly detectors while the latter is known as signature detectors. In Bishop (2003), an anomaly detector has a set of activities or profiles to represent "normal behaviors" in its detection engine. Operators of the IDS can derive normal behaviors from the historical behaviors of the host, operating system, application and the users of the networks. The detector then compares inbound and outbound traffics with its profiles and subsequently raises alerts for traffics deviate from the normal behaviors. The significance of this design is its capability to detect new attacks. However, the major concern about anomaly detectors is the integrity of the reports they generate. Secondly, activities that constitute normal and abnormal behaviors can change over time (Chandola and Kumar, 2009).

Misuse detectors are also called signature-based detectors because they keep databases of patterns, known vulnerabilities or signatures of known and anticipated attacks (Bishop, 2003; Wang et al. 2006).

The IDS that uses misuse detection methods usually compares incoming and outgoing traffics with each of its detection rules in a top-down manner. The detector will subsequently trigger alerts whenever a packet matches any of its rules to indicate the presence of suspicious message intending to access the computer. Conversely, the mechanism will ignore a packet that does not match any of its rules by treating each of them as a normal packet (Bishop, 2003). However, a signature-based detector can only detect attacks that match its detection rules.

Most signature-based detectors are criticised for the inability to decode encrypted traffics (Scarfone and Mell, 2007). Network intrusion detectors have limited capacity to process packets. For this reason, some of them can drop significant number of packets whenever attackers overload them with network traffics.

In effect, misuse and anomalous IDSs have several flaws. Operators must constantly update profiles of anomaly detectors and the signatures of misuse detectors (Karthikeyan and Indra, 2010).

*d) Classification by intervals between detection and analysis*

In Lazarevic et al. (2005), IDSs are classified into real-time and off-line systems. A real-time intrusion detector analyzes computer activities while in progress and concurrently raises alerts once an attack is detected. Contrarily, off-line intrusion detector reports activities after the events have happened.

Furthermore, giving the inadequacies of detection capacities of the current versions of IDSs, it is plausible that analyzers of intrusion logs can take wrong decisions against legitimate events in a real-time manner.

Similarly, an off-line intrusion detection mode exposes computer resources to risks, especially if there is a relatively long time interval between the time the detector detects the attacks and the time to review the IDS logs.

*e) Classification by method of deployment*

There are centralized, distributed and hybrid intrusion detection models (Lazarevic et al. 2005). A centralized IDS usually aggregates alerts of other IDSs at a fixed location. The detector can easily detect stealthy attacks that below threshold operators have defined in each segment of the network whenever they analyze intrusion logs at a central location.

Nevertheless, the efficacy of this design depends on stable communications between the contributing sources and the repository where the operators will analyze the data. Furthermore, the capability of centralized IDS to overcome discrepancies that may exist within the logs of different models of IDS is another weakness that is peculiar to this model.

Distributed intrusion detectors analyze logs of computer activities in individual locations. In Debar et al. (2000), the benefit of this model is that multiple intrusion logs can be used to validate each other in reducing false positives. Nevertheless, security experts often encounter different challenges whenever they have to review several intrusion logs.

Also, a hybrid model combines centralized and distributed models to achieve high intrusion detection rate. Nonetheless, integrated IDSs often combine the weaknesses inherent in all the cooperating IDSs.

*f) Classification by method of reporting*

The action that an IDS takes upon the detection of an intrusion has a significant impact on the group the detector belongs. Hence, Lazarevic et al. (2005) group IDSs into passive and active response models. The passive response detectors can not deter attacks in progress, unlike the active response detectors that can generate alerts and initiate preventive actions to block attacks from achieving the objectives of the attackers. The major problem with passive and active response models is that both approaches still exhibit shortcomings that are similar to that of the real-time and offline models (Lazarevic et al. 2005).

The fundamental truth is that all the above models of IDS collectively generate alerts such as shown in Figure 3 and such information can degenerate to series of problems.

## III. Emerging Issues with Formats of IDS Alerts

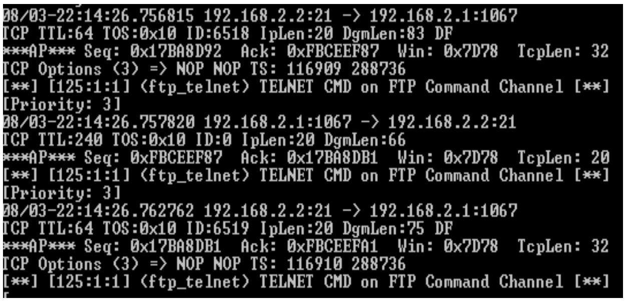
IDSs organize, log and display alerts in different manner. This paper uses Bro and Snort IDS as examples of NIDSs (Alder et al. 2007; Bro, 2017). For instance, Snort logs alert in ASCII and full alert's formats. Nonetheless, ASCII formats cannot be immediately discernible or readable by human operators. Operators will still need specialized tools to decode, read and analyze them before they can make meanings decisions from them. This indicates a danger if the analyzers that can decode the logs are not readily available and operators must promptly take decisons to discern suitable countermeasures that will thwart attacks signified by such logs.

Snort can generate comprehensive information that will include the packet's headers and Snort's assigned attributes. The mechanism can further assign the rule that triggers the alerts, the description, time and date the event is logged. The detector can be configured to produce different output modes such as fast, full or console. This functionality enables the operators to configure Snort to generate less output whenever such requirements arise.

Each NIDS has its peculiar signatures and formats for writing the detection rules. For example, Bro captures comprehensive information about suspicious traffics into tab-separated log files. Such verbose narrations usually include each the host, connection, extraction of vital information from many application-layer protocols and server responses. The major strengths of NIDSs are many. Experience suggests that NIDSs such as Snort and Bro can analyze PCAP files in offline mode and IPv4 and IPv6 formats (Bro, 2017). The detectors can be used for forensic analysis of intrusive evidence in real-life networks.

## IV. Emerging Issues with Kinds of IDS Alerts

Existing IDSs trigger "disused alerts" and "used alerts". The former are categories of warnings that analysts will never use for any significant purpose. Also, they are warnings that are mostly abandoned by professionals for some reasons. However, it is usually hard to establish the degree of severity of such messages without making a thorough investigation about them. Hence, analysts must be prudent in handling them in a realistic environment.



```
08/03-22:14:26.756815 192.168.2.2:21 -> 192.168.2.1:1067
TCP TTL:64 TOS:0x10 ID:6518 IpLen:20 DgmLen:83 DF
***AP*** Seq: 0x17BA8D92  Ack: 0xFBCEEF87  Win: 0x7D78  TcpLen: 32
TCP Options (3) => NOP NOP TS: 116909 288736
[**] [125:1:1] (ftp_telnet) TELNET CMD on FTP Command Channel [**]
[Priority: 3]
08/03-22:14:26.757820 192.168.2.1:1067 -> 192.168.2.2:21
TCP TTL:240 TOS:0x10 ID:0 IpLen:20 DgmLen:66
***AP*** Seq: 0xFBCEEF87  Ack: 0x17BA8DB1  Win: 0x7D78  TcpLen: 20
[**] [125:1:1] (ftp_telnet) TELNET CMD on FTP Command Channel [**]
[Priority: 3]
08/03-22:14:26.762762 192.168.2.2:21 -> 192.168.2.1:1067
TCP TTL:64 TOS:0x10 ID:6519 IpLen:20 DgmLen:75 DF
***AP*** Seq: 0x17BA8DB1  Ack: 0xFBCEEFA1  Win: 0x7D78  TcpLen: 32
TCP Options (3) => NOP NOP TS: 116910 288736
[**] [125:1:1] (ftp_telnet) TELNET CMD on FTP Command Channel [**]
```

*Figure 3:* Snort's alerts on a publicly available dataset

Conversely, the latter are warnings that analysts use for decision purposes such as the investigation of the incident of intrusions, designing countermeasures and mitigation's strategies. Redundant warnings, alerts workload and diverse processing methods for processing IDS alerts are central aspects of emerging issues associated with "used alerts" that are within IDS logs in a recent time.

### a) Redundant alerts

Redundant alerts are fundamental problems of intrusion detection technology. These issues are the main challenges to the usage of IDSs for network forensics over the years because they can complicate the problems of classification, data reduction, false positive; intrusion correlation and reporting (Nehinbe, 2011; Tjhai et al. 2008).

It is possible to explain the above concept in three different perspectives: The first problem is how to reasonably reduce the entire alerts in an intrusion log without underestimating security breach the IDS has reported (Nehinbe, 2011). The second challenge is how to promptly discern false warnings from realistic attacks so that operators will not implement countermeasures are against legitimate events (Stallings, 2011). The third issue is how to eliminate less critical alerts from an intrusion log to enhance clarity of the reports.

Redundant alerts originate from the point at which the NIDS decides on the network packets that it would respectively classify as suspicious and normal packets or activities (Scarfone and Mell, 2007). On the whole, every NIDS has detection rules or signatures, patterns or characteristics of events that suggest intrusions. The detector uses the rules to validate each of the packets that the detector notices.

Fundamentally, the detector will raise an alert each time a packet matches its detection rule to signify an intrusion or suspicious activity. The mechanism records the warnings inside the log in the order of occurrence for further review. NIDS treats outbound or inbound traffic as a new occurrence within the same timestamp. Hence, the IDS toolkit often triggers overwhelming alerts that may suggest notices of closely related packets (Nehinbe, 2010). Therefore, analysts automatically inherit the classification problems that the detector cannot adequately tackle.

### b) *Alerts workload*

Human operators must re-examine the content of IDS logs. Usually, more time and efforts are spent to ascertain the correctness of the redundant warnings, and to substantiate suitable preventive measures. Furthermore, the occurrence of indiscernible relationships among the entries within the log can complicate the process of analyzing them.

Furthermore, the problems of alerts workload can degenerate to swamping whereby the detector triggers excessive warnings that exceed the capability of the analyst. One of the established approaches to lessen the problems of alerts workload is to configure the detector to suppress some significant quantity of alerts at a specified time and by ignoring specific network traffic (Alder et al. 2007; Rehman, 2003; Scarfone and Mell, 2007). Similarly, operators can configure the detectors to trigger specific quantity of alerts. The operators can also deactivate nuisance rules. Also, they can reconfigure the IDS by prioritizing the detection rules so that rules that have low priorities will trigger little or no alerts. Nevertheless, any of the methods above will only be possible to be carried out with a detector that has such functionalities.

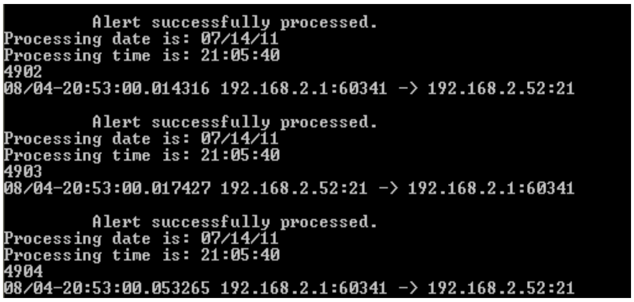
Secondly, alerts suppression techniques are vulnerable to the high rate of false negatives, especially whenever an intruder attacks a target machine with probing attacks that are below the threshold for suppressing the alerts. For instance, a packet of ping attack that is below the threshold is enough to evade detections.

Alerts suppression techniques have a propensity to bury small relationships that are sneaky intruders deliberately embedded in multiple alerts. For these reasons, alerts suppression methods frequently underestimate security breaches on the computer and mobile networks.

Moreover, it is cumbersome to reconfigure all the detection rules that NIDS uses as a method for reducing alerts workload (Alder et al. 2007). These tradeoffs have necessitated the implementation of NIDS in a default mode while operators can decide to adopt correlation and aggregation techniques to manage the problems of alerts workload that are inherent in its operations.

### c) *Different methods for processing IDS alerts*

There are numerous ways and approaches to process alerts logged by IDSs. For instance, Figure 4 shows how we analyze alerts from Snort in the course of implementing clustering of intrusive trace files by C++ programs.



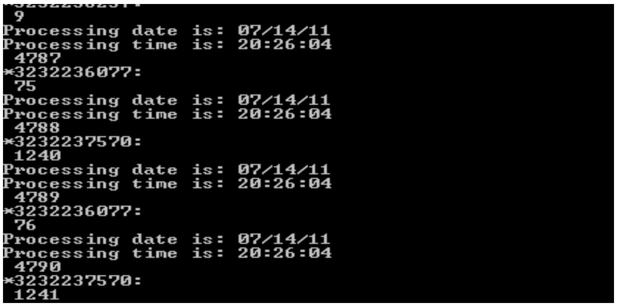
```
        Alert successfully processed.
Processing date is: 07/14/11
Processing time is: 21:05:40
4902
08/04-20:53:00.014316 192.168.2.1:60341 -> 192.168.2.52:21

        Alert successfully processed.
Processing date is: 07/14/11
Processing time is: 21:05:40
4903
08/04-20:53:00.017427 192.168.2.52:21 -> 192.168.2.1:60341

        Alert successfully processed.
Processing date is: 07/14/11
Processing time is: 21:05:40
4904
08/04-20:53:00.053265 192.168.2.1:60341 -> 192.168.2.52:21
```

*Figure 4:* Processing alerts from Snort

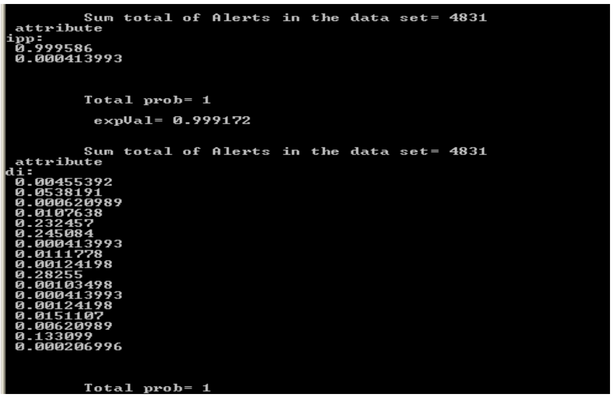
In Nehinbe (2011), some authors have used Neural Networks (NN), Genetic programming, Visualizations; and Petri net to analyze the same category of publicly available datasets for testing IDS models in a different context (Wang et al. 2006).



```
 9
Processing date is: 07/14/11
Processing time is: 20:26:04
 4787
*3232236077:
 75
Processing date is: 07/14/11
Processing time is: 20:26:04
 4788
*3232237570:
 1240
Processing date is: 07/14/11
Processing time is: 20:26:04
 4789
*3232236077:
 76
Processing date is: 07/14/11
Processing time is: 20:26:04
 4790
*3232237570:
 1241
```

*Figure 5:* Alerts from Snort

Similarly, analysts can adapt the same group of alerts from the IDS such as Snort IDS for different purposes. For examples, Figure 5 illustrates how timestamp can be used to group alerts from Snort on the trace files into different clusters while Figure 6 gives the statistical transformation we carried out with the same trace file.



```
        Sum total of Alerts in the data set= 4831
 attribute
ipp:
 0.999586
 0.000413993


        Total prob= 1
         expVal= 0.999172

        Sum total of Alerts in the data set= 4831
 attribute
di:
 0.00455392
 0.0538191
 0.000620989
 0.0107638
 0.232457
 0.245084
 0.000413993
 0.0111778
 0.00124198
 0.28255
 0.00103498
 0.000413993
 0.00124198
 0.0151107
 0.00620989
 0.133099
 0.000206996


        Total prob= 1
```

*Figure 6:* Statistical analysis of logs of Snort

Some authors have used other programming languages to process the same public trace files and to achieve different objectives. The central problem here is that it is difficult to substantiate which of the available methods and programming languages for analyzing logs of IDSs are the best ways to present such events in the context of digital security and forensics.

## V. Conclusion

The possibility that victims of intrusions can suffer serious loss of business and trade secrets is a major concern across the globe. This paper critically reviews the evolution of the IDSs since the 1980s and some technical issues that arise with the existing models over the years. Thus, we also discuss a wide range of taxonomy together with their strengths and weaknesses.

Furthermore, we examine potential loss that victims of intrusions can experience. We affirm that intrusions can modify and delete a listing of the files stored in the memory of a computer system. Intrusions can embarrass private users and corporate firms. Intruders can divulge classified information about the governments, agencies, corporate firms and highly dignified people to their competitors, opponents and enemies.

Also, we show that there are overheads regarding control, spending, cost, apportioning of resources and the mechanisms necessary to quickly thwart intrusions in a real-life environment. However, series of technical issues were erroneously over-sighted over the years. This paper thoroughly presents a new review of the IDS technology to lessen them.

Overview of the weaknesses of IDSs collectively suggests that they can trigger many redundant alerts. Such alerts can degenerate to the problems of swamping if the trade-offs between true positives and false positives are not methodologically balanced. Hence, a thorough review of intrusion log requires a high level of expertise to establish the meaning and validity of each alert.

Furthermore, capabilities of attributes of alerts in the intrusion logs to discriminate attacks are some of the emerging issues we have mentioned above. The vast majority of the models we have reviewed above must be evaluated across a wide range of synthetic and realistic datasets. They must also be evaluated with big datasets to establish their performances with large and small evaluative datasets.

Additionally, intrusion aggregation techniques lack the capability for detecting patterns of attacks because they are unable to isolate alerts that respond to failed packets from suspicious activities that can reach their destinations.

Some intrusion aggregation models fundamentally reduce alerts redundancies and workload by focusing only on alerts with high priorities. Hence, suspicious activities that have low priorities may easily elude detections.

The underpinning theories and principles of some research designs may not be very useful for solving real-world problems. Graphical approaches usually produce series of hyper-alerts and numerous correlation graphs with numerous nodes. Graphical approaches tend to produce edges that are difficult to interpret.

Above all, the review above has not described how IDSs can eliminate ineffectiveness and inability to discriminate alerts by the information content they convey. We have not discussed existing mechanisms that are designed to ensure the predictability of each attribute IDSs extracted to describe suspicious packets. These are areas of further research direction that can be pursued to reduce the issues above and to improve the efficacies of IDSs in general.

## References Références Referencias

1. Alder, R., Baker, A.R., Carter, E.F., Esler, J., Foster, J.C., Jonkman, M., Keefer, C., Marty, R. and Seagren, E.S. Snort: IDS and IPS Toolkit, Syngress publishing, Burlington, Canada, 2007.
2. Axelsson, S. Intrusion Detection Systems: A Survey and Taxonomy, Department of Computer Engineering, Chalmers University of Technology, Goteborg, Sweden, 2000.
3. Amer. S.H. and Hamilton, J.A. Intrusion Detection Systems (IDS) Taxonomy – A short review, 2011.
4. Anderson, J.P. Computer Security Threat Monitoring and Surveillance, Technical Report Washing, PA, James P. Anderson Co., 1980.
5. Bishop, M. Computer Security: Art and Science, Pearson Education, Inc, New York, 2003.
6. Gary, M. (2007), Silver Bullet Talks with Becky Bace, IEEE Security & Privacy Magazine, Vol. 5 , pp. 6–9.
7. Bro (2017), Bro Logging; https://www.bro.org/documentation/index.html; Accessed 14/12/2017
8. Chandola, V. and Kumar. V. Anomaly detection: A survey, University of Minnesota, 2009.
9. Debar, H., Dacier, M. and and Wespi, A. A Revised Taxonomy for Intrusion-Detection Systems, Annals of Telecommunications, vol. 55, pp. 361-78, 2000.
10. Ghorbani, A.A., Lu, W. and Tavallaee, M. Network Intrusion Detection and Prevention: Concepts and Techniques, Springer, New York, LLCC, 2010.
11. Kizza, J.M. A Guide to Computer Network Security, Springer-Verlag London, 2009.
12. Karthikeyan .K.R. and Indra, A. Intrusion Detection Tools and Techniques- A Survey, International Journal of Computer Theory and Engineering, Vol. 2, pp. 901-906, 2010.
13. Lazarevic, A., Kumar, V. and Srivastava, J. Intrusion detection: A survey, Managing Cyber Threats, pp. 19–78, June 2005.
14. Mohamed, A.A. (2013), Design Intrusion Detection System Based On Image Block Matching, International Journal of Computer and Communication Engineering, Vol. 2.

15. Nehinbe, J.O. Methods for reducing workload during investigations of intrusion logs, PhD thesis, University of Essex, UK, 2011.
16. Nehinbe J.O. Concurrent Reduction of False Positives and Redundant Alerts, International Conference on Information Society (i-Society 2010), proceedings of IEEE, London, UK.
17. Rehman, R. Intrusion Detection Systems with Snort: Advanced IDS Techniques Using Snort, Apache, MySQL, PHP and ACID, Prentice Hall PTR Upper Saddle River, New Jersey, 2003.
18. Stavroulakis, P. and M. Stamp. Handbook of Information and Communication Security, Springer-Heidelberg, Dordrecht London New York, 2010.
19. Scarfone, K. and Mell, P. Guide to Intrusion Detection and Prevention Systems (IDPS), Recommendations of the National Institute of Standards and Technology, Special Publication 800-94, Technology Administration, Department of Commerce, USA, 2007.
20. Stallings, W. Network Security Essentials: Applications and Standards, 4th edition, Prentice Hall, 2011.
21. Tjhai, G.C., Papadaki, M., Furnell, S.M. and Clarke, N.L. The Problem of False Alarms: Evaluation with Snort and DARPA 1999 Dataset, Trust, Privacy and Security in Digital Business, LNCS Vol. 5185, pp. 139–150, Springer-Verlag Berlin Heidelberg, 2008.
22. Wang, J., Wang, Z. and Kui-Dai. Intrusion Alert Analysis Based on PCA and the LVQ Neural Network, Neural Information Processing Lecture Notes in Computer Science, Vol. 4234, pp. 217-224, 2006.

Global Journals Inc. (US) Guidelines Handbook 2017

www.GlobalJournals.org