Journal of Computer Networks, 2017, Vol. 4, No. 1, 48-55
Available online at http://pubs.sciepub.com/jcn/4/1/5
©Science and Education Publishing
DOI:10.12691/jcn-4-1-5

Big Data in Intrusion Detection Systems and Intrusion Prevention Systems

Lidong Wang*

Department of Engineering Technology, Mississippi Valley State University, Itta Bena, MS, USA
*Corresponding author: lwang22@students.tntech.edu

Abstract This paper introduces network attacks, intrusion detection systems, intrusion prevention systems, and intrusion detection methods including signature-based detection and anomaly-based detection. Intrusion detection/prevention system (ID/PS) methods are compared. Some data mining and machine learning methods and their applications in intrusion detection are introduced. Big data in intrusion detection systems and Big Data analytics for huge volume of data, heterogeneous features, and real-time stream processing are presented. Challenges of intrusion detection systems and challenges posed by stream processing of big data in the systems are also discussed.

Keywords: *big data, intrusion detection system (IDS), intrusion prevention system (IPS), signature-based detection, anomaly-based detection, data mining, machine learning, network security*

Cite This Article: Lidong Wang, “Big Data in Intrusion Detection Systems and Intrusion Prevention Systems.” *Journal of Computer Networks*, vol. 4, no. 1 (2017): 48-55. doi: 10.12691/jcn-4-1-5.

1. Introduction

Many classes and applications of cybercrime and terrorism contain a misrepresentation of identity or an attempt to authenticate for access to a business or services for which attackers have no legitimate use. Within the European Union, the eIDentity, Authentication & Signatures Regulation were launched in October 2014. The initial results of the European project CAMINO in terms of the realistic roadmap to counter cybercrime and cyber terrorism were presented. The primary target for the CAMINO project was to provide a realistic roadmap for improving resilience against cyber terrorism and cybercrime [1]. An intrusion detection system (IDS) is often regarded as a second-line security solution after authentication, firewall, cryptography, and authorization techniques, etc. which are first line security measures [2]. An IDS is software that automates the intrusion detection process. An intrusion prevention system (IPS) is software that has all the capabilities of an IDS and can also attempt to stop possible incidents. IDS and IPS technologies can offer many of the same capabilities, but administrators can also disable prevention features in IPS products, letting them function as IDSs. Many intrusion detection and prevention systems (IDPS) can also respond to a detected threat and use several response techniques: during which the IDPS can stop the attack itself, change the attack’s content, or change the security environment (e.g., reconfiguring a firewall) [3].

An IDS can monitor specific protocols like the Hyper Text Transfer Protocol (HTTP) of a web server. This type of IDS is called a protocol-based intrusion detection system (PIDS). IDSs can also be specialized to monitor application-specific protocols like an application protocol-based intrusion detection system (APIDS). An example of this is an APIDS, which monitors the database’s Structured Query Language (SQL) protocol. Like the heterogeneity of the security event sources such as network and diverse host types, the IDSs themselves can also be heterogeneous in their types, how they operate, and their diverse alert-output formats [4].

Four kinds of data can be gathered for correlation by a developed IDS in security monitoring. They are: IP flow records, HTTP packets, DNS replies, and Honeypot data. For example, flow records provide invaluable data for detecting intrusions or highlighting botnet communications. Traces of every communication from the enterprise network to the Internet and vis versa could be stored by exporting NetFlow records from the core router of the network. HTTP traffic is a well-known intrusion vector and represents a significant portion of the traffic of Internet users. Studying uniform resource identifiers (URIs) embedded in HTTP packets and their payload help detect and prevent malicious communications. Domain Name System (DNS) requests are performed to get IP addresses associated with a domain and consult the associated resource. Therefore, monitoring the DNS to identify malicious domains is efficient in proactively detecting and preventing an important part of malicious communications. A honeypot generally emulates vulnerable services and contains fake production data. Logging honeypot information helps obtain attackers’ data about targeting a specific network such as protocols used, IP addresses used, exploit file used, and scanning strategies, etc. [5].

There are three models of intrusion detection mechanisms: signature-based, anomaly-based, and hybrid detection [6]. However, two approaches of attack identification are usually used in an IDS: 1) signatures that are specific defined elements of the network traffic and are possibly useful for identification; and 2) anomalies that are some deviation of the normal network behaviour. In the above both situations, one must pre-define the form of the signature and the network's normal behaviour [7]. Signature detection is also called misuse detection.

This paper focuses on the following aspects: 1) attacks and intrusion detection methods including IDPS and attacks, signature-based detection, anomaly-based detection, and the challenges of intrusion detection systems; 2) some data mining and machine learning methods used in intrusion detection systems; 3) big data in intrusion detection systems including huge volumes of data and data fusion for heterogeneous sources, and real-time stream data and big data stream processing.

2. Attacks and Intrusion Detection Methods

2.1. Intrusion Detection and Prevention Systems (IDPS) and Attacks

Attacks can be divided into the following four main categories [2,7,8]: 1) denial of service (DoS) — an attacker tries to prevent legitimate users from using a service; 2) probe — an attacker tries to find information about the target host through ways such as scanning victims to get information about available services and the operating system; 3) U2R (user to root) — unauthorized access to local superuser (root) privileges; and 4) R2L (remote to local) — unauthorized access from a remote machine through approaches such as guessing password to obtain a local account on the victim host.

An advanced persistent threat (APT) is a targeted attack against a high-value asset or physical system. APT attackers often leverage stolen user credentials or zero-day exploits to avoid triggering alerts. Thus, this type of attack can take place over an extended period of time while the victim organization remains oblivious to the intrusion. Existing anomaly detection methods commonly focus on obvious outliers (e.g., volume-based); but are ill-suited for stealthy APT attacks, thus suffering from high false positive rates. Since APT attacks consist of multiple stages, each action by the attacker provides an opportunity to detect behavioural deviations from the norm. Correlating these seemingly independent events can reveal evidence of the intrusion and expose stealthy attacks [9].

Table 1. The Classification of Intrusion Detection Systems

Signature/Misuse Detection	Anomaly Detection
○ Programmed	○ Programmed
□ Expert System	□ Default Deny
□ String Matching	• State Series Modeling
□ Simple Rule Based	□ Descriptive Statistics
□ State Modelling	• Simple State
• Petri Net	• Threshold
• State Transition	• Simple Rule Based
	○ Self-Learning
	□ Time Series
	• Artificial Neural Network
	□ Non-Time Series
	• Rule Modeling
	• Descriptive Statistics

Intrusion detection approaches can be classified into five subcategories: statistics, rule, state, pattern and heuristic based. It is concluded that the pattern-based approach is effective in identifying unknown and hidden attacks [10]. Table 1 [10] shows the IDS classification.

Attackers sometimes are willing to spread their actions over a wide period to evade detection systems. Therefore, it is necessary in this situation to shift the focus away from real-time detection which significantly limits analysis and correlation capabilities. Instead, an approach focused on full-packet capture, deep packet inspection and Big Data analytics that enable to use more advanced algorithms for analysis and correlation and mitigating such evasion attempts is preferable. Although offline analysis (analysis of captured traffic) inevitably results in delayed attack detection, it is imperative to consider that perpetrators sometimes spend a significant amount of time trying to reach a specific objective (e.g., exfiltrate sensitive data) in the majority of APTs [11].

Organizations should consider using a system with multiple types of IDPS technologies to achieve more accurate and comprehensive performance in the detection and prevention of malicious activity. The four primary types of IDPS technologies include host., network, network behaviour analysis (NBA)-based, and wireless-based; each offers different logging, information gathering, detection, and prevention capabilities [3]. Gnort, an intrusion detection system that uses the GPU to offload pattern matching computation was introduced. The system is based on the Snort open-source NIDS (network intrusion detection system) that exploits the underutilized computational power of modern graphics cards to offload the costly pattern matching operations from the CPU, thus increasing the over-all processing throughput. Gnort has achieved a maximum traffic processing throughput of 2.3 Gbit/s using synthetic network traces, while using a commodity Ethernet interface when monitoring the real traffic. The results suggest that modern graphics cards can be used effectively to speed up intrusion detection systems as well as other systems that involve pattern matching operations [12].

2.2. Signature-based Detection

Predefined attack specifications have to be provided to an IDS for misuse (signature) detection, which requires human security experts to manually analyse attack related data and formulate attack specifications. Attack specifications can be generated automatically by using various automated techniques. However, most of the misuse detection systems lack this capability and most of the systems focus on data produced by single source [13]. There are four categories of signatures [14]:

- String signatures: The string signature engines support regular expression pattern matching and alarm functionality.
- Connection signatures: They generate an alarm based on the conformity and validity of the network connections and protocols.
- DoS signatures: They contain behavior descriptions that are considered characteristics of a DoS attack.
- Exploit signatures: They typically identify a traffic pattern that is unique to a specific exploit; therefore,

each exploit variant may require an individual signature. Attackers may be able to bypass detection by slightly modifying the attack payload. One often must produce an exploit signature for each attack tool variant.

Current anti-virus solutions are vulnerable to zero-day attacks because they are signature based and anomaly-based detection lacks the reliable mechanism to construct an accurate profile to distinguish the attacks from normal events. Although it is extremely difficult to develop an effective solution for defending all unknown attacks, it has been found that most of them have one thing in common — hidden executable content. An anomaly based solution was proposed to detect hidden executable content in the network traffic. Anti-virus products belong to the signature-based detection approach which identifies threats by matching known features. This method provides high accuracy for attacks already known, but not effective for zero-day attacks. Zero-day attacks include new type threats and the variations of existing attacks which have no evidence of specific features at their first launches. The only solution to defend against zero-day attacks is the anomaly-based detection independent of specific signatures [15].

2.3. Anomaly-based Detection

Anomaly based detection is also called "behaviour-based detection". It is an IDS method which models the behaviour of the network, users, and computer systems and raises an alarm whenever there is a deviation from the normal behaviour. It is particularly good at identifying probes and sweeps towards network hardware. It can give early warnings of potential intrusions because scans and probes are the predecessors of all attacks [16]. Anomaly detection may be divided into static and dynamic anomaly detection. Static anomaly detectors usually only address the software portion of a system; therefore, they focus on integrity checking. Dynamic anomaly detection typically operates on audit records or monitored networked traffic data [13].

Detectors of behavioural deviations are referred to as “anomaly sensors” with each sensor examining one aspect of the host’s or user’s activities within a network. For instance, a sensor may profile the set of machines that each user logs into to find anomalous access patterns; keep track of the external sites that a host contacts to identify unusual connections; study users’ regular working hours to flag suspicious activities in the middle of the night; or track the flow of data between internal hosts to find unusual “sinks” where large amounts of data are gathered. The triggering of multiple sensors suggests more suspicious behaviour [9]. Much research has focused on the network traffic in the flow-level network. Therefore, a novel approach was proposed to reveal the abnormal patterns by dealing with the packet-level network data based on the latest method from compressed sensing [17].

Since current Internet threats include not only malicious codes like Trojan or worms, but also spyware and adware which do not have explicit illegal content. It is necessary to have a mechanism to prevent downloading hidden executable files in the network traffic. A solution was presented to identify executable content for the anomaly-based network intrusion detection system (NIDS) based on the file byte frequency distribution [15]. One way to glean more context in network traffic analysis is to investigate what kind of data is being moved. With deep packet inspection and session reassembly, one can perform file-based analysis of content for improving detection accuracy and compare current traffic with baselines to look for anomalies in data movement. The following situations help identify anomalies [18]:

- Time of day: If a user doesn’t normally work in the middle of the night, but he or she does so two days in a row, this could indicate malicious activity.
- Application patterns: Many web-based applications have known and predictable transaction patterns that can be profiled. One can look for anomalies according to an established baseline.
- File size: If a user moves, for example, 2GB of traffic in 24 hours, but he or she normally move no more than 100MB per day, an alert should be triggered.
- Simple DLP: One can fingerprint files to look for sensitive content or regular expressions which match account numbers or other protected data. This isn’t full DLP classification and analysis, but it could flag something as malicious if there is no overhead of full DLP.

In recent years, sampled traffic data has also been used as input for anomaly detection, e.g., detecting DoS attacks or worm scans. It is well known that sampling distorts traffic statistics such as mean rate and flow size distribution. Two kinds of sampling have been widely discussed in literature. They are: flow sampling and packet sampling. Packet sampling is simple to implement with low CPU power and memory requirements. Flow sampling emerges as an alternative to overcome the limitations of packet sampling. It is shown to improve accuracy but still suffers from prohibitive memory and CPU power requirements. Packet traces captured from a Tier-1 IP-backbone were sampled using four popular methods: random flow sampling, random packet sampling, smart sampling, and sample-and-hold. The sampled data is then used as input to detect two common classes of anomalies: port scans and volume anomalies. Port scanning is usually associated with worm or virus propagation, while volume anomalies can occur due to a variety of reasons, including flash crowds and DoS attacks. Several port-scan detection techniques have been proposed. For instance, SPICE performs a complex off-line Bayesian analysis to detect stealthy port scans. Snort is a flexible open-source intrusion detection system that issues scan alerts based on user-defined connection patterns and rates. Threshold Random Walk (TRW) and Time Access Pattern Scheme (TAPS), two effective “on-line” port-scan detection techniques, were presented. Three representative algorithms including a wavelet-based volume anomaly detection and two port-scan detection algorithms were studied based on hypotheses testing [19]. Table 2 [20] shows the mapping of attacks with anomalies. Point anomalies corresponds to data samples which qualify as anomaly with respect to the rest of the dataset. Contextual anomaly (also called conditional anomaly) refers to an anomalous behaviour which is considered as an anomaly only in certain contexts and not in others. Collective anomalies refer to the collections of data samples which are anomalous altogether [21,22].

Table 2. Mapping of Various Attacks with Anomalies

Anomaly	Network Attacks
Point	U2R, R2L
Contextual	Probe
Collective	DoS

2.4. Comparison of Intrusion Detection/Prevention System (ID/PS) Methods

Compared with signature-based approaches, anomaly-based approaches can lead to a faster execution, but often result in a high false positive rate. Since the intrusion patterns and normal patterns do not always comply with certain distributions, nor are they linearly separable; therefore, this situation causes problems when applying statistical learning methods such as support vector machine (SVM) for intrusion detection [23]. Ignorance of the packet payload is a major reason for the poor performance of anomaly detection on the application level. In signature-based methods, intense analysis has been made on the packet payload to extract the unique signatures, so the signature-based approach can provide extremely high accuracy on existing attacks [15]. Table 3 [24] compares ID/PS methods and their features.

Table 3. Comparison of ID/PS Methods and Features

Aspects	Methods	Advantages	Disadvantages
Audit source location	Host based	• Do not require additional hardware. • Cost effective. • Easy to deploy due to not affecting existing infrastructures. • Able to see low-level local activities such as file accesses, changes to file permissions. • Can deal with encrypted and switched environments.	• A very limited view of the network. • More disposed to illegal tampering due to being close to users.
	Network based	• Quick response. • Less prone to false positives than host-based systems. • Able to detect attacks missed by host-based systems due to monitoring network traffic at the transport layer.	• Not be aware of implementation of each host's protocol due to being far from individual hosts. • No capability to decrypt encrypted data. • Difficulty to remove evidence.
Detection methods	Anomaly	• Able to detect most new attacks. • Using fewer rules compared with the signature based techniques, thus increasing detection rate and effectiveness.	• Difficult to discover the boundaries between abnormal and normal behavior. • Higher false positive alarms. • Difficulty in adapting to continuously changing normal behavior and dynamic anomaly.
	Misuse	• Be reliable, efficient, and generate a very low false alarm rate in detecting specified and well-known intrusions.	• False alarms due to poorly constituted signatures. • Limitation in unknown attacks. • Matching signatures is well done for single connection attacks only, while most of the attacks involve multiple connections.
Data distribution modes	Central	• All of the monitoring, detection, and response activities are controlled directly by a central console.	• Data can be destroyed or modified by an attacker. • An intruder can modify or disable the programs running on a system
	Distributed	• The distributed data utilizes the traffic information from various sources to investigate the security status.	• The data flow between host monitors and the director agent may generate high network traffic overheads.
Technology layout	Wired	• Wired networks are faster and low cost.	• Heavily dependent on structure platform and not easy to deploy.
	Wireless	• Wide coverage and unlimited access. • Mobile agents have less energy consumption. • Scalable and independent from infrastructure platform.	• The wireless medium itself has to be protected in addition to attacks that may be performed on a wired network.
Time of detection	Non real-time	• Less resource consumption. • High capabilities to provide the evidence of data forensic.	• Cannot provide real time response to prevent or mitigate damages.
	Real-time	• Excels in attack detection and prevention. • Can fill the network inherent security gaps associated with vulnerability to various types of attacks (especially DoS) that are not detectable by common approaches.	• The performance is affected by a running agent through the system. • Cannot handle encrypted packets. • The source address can be spoofed and makes it hard to trace and responds attacks automatically.

2.5. Challenges of Intrusion Detection Systems

One challenge for IDS devices deployed over a large network lies in that IDS components communicate across sub-networks, sometimes through firewalls and gateways. For different parts of the network, network devices may use different data formats and different protocols for communication. The IDS must be able to recognize the different formats. Another challenge for the IDS in a large network is effectively monitoring traffic. Network intrusion detection system (NIDS) components are scattered throughout a network. If the components are not placed strategically, many attacks can bypass NIDS sensors by traversing alternate paths in a network [25]. A major challenge for current IDSs is the limited time window for which the connection state can be maintained. As all modern IDSs are focused on real-time detection, they can only support a short time window (usually a few seconds) in which attacks can be detected for Transmission Control Protocol (TCP) sessions. Port scanning is a practical example of this weakness. A quick port scan against a host will trigger an alert from virtually any IDS. However, if this scan is spread over a period of several minutes, the attack will pass undetected for the majority of IDSs [11]. In networks, the following challenges are still open issues and summarized below for intrusion detection solutions [26]:

- Runtime limitation: Real time intrusion detection should capture and inspect each packet without any packet loss. High traffic load can impact the capture and inspection methods and requires a power solution to this issue.
- Number of false positives: Reducing computational complexity in pre-processing is required.
- The intrusion detection setup should be independent of its infrastructure.
- Attack anomalies changes and going undetected: Detection profiles and methods should be dynamically updated and adapted in order to detect new attack patterns without compromising the performance.

3. Some Data Mining and Machine Learning Methods Used in Intrusion Detection

The research performed from the years 2001 till 2008 was summarized and the outcome showed that the attack detection research emphasized trying to find hybrid solutions and detection classification. However, the outcome of the research performed from the years 2010 till 2015 showed that the attack detection research during this period emphasized more on machine learning and data mining including hybrid solutions with misuse-based and anomaly-based intrusion detection [26].

The analysis of stream data is important. Due to the transient and dynamic nature of intrusions and malicious attacks, it is necessary to perform intrusion detection in the data stream environment. Furthermore, an event may be normal on its own, but it is considered malicious if it is viewed as part of a sequence of events. Therefore, it is necessary to study what sequences of events are frequently encountered together, find sequential patterns, and identify outliers. Data mining methods for finding evolving clusters and building dynamic classification models in data streams are also necessary for real-time intrusion detection [27]. Outlier detection is an instance of data mining and it is useful for intrusion detection. There are two types of outliers. The first type is the one that deviate significantly from others within their own network peripherals, while the second type is the one whose patterns belongs to other network services other than their own service. Much research has shown that it is extremely difficult to find out outliers directly from high dimensional datasets; therefore, much work has been done in reducing the dimensionality of the dataset. Dimension reduction can be performed by principal component analysis (PCA) [23].

Intrusions can be launched from several different locations and target to many different destinations. Distributed data mining methods may be used to analyse network data from several network locations to detect these distributed attacks [27]. Evolving data stream mining classifiers have been used in massive online analysis (MOA) as they are capable to handle concept drift in data streams. There are 16 evolving data stream classifiers in MOA [2]. Unsupervised intrusion detection can be based on clustering that aims to group data instances together into clusters. All the instances that appear in small clusters are labelled as anomalies because normal instances should form large clusters compared to intrusions [28]. Table 4 [20] shows a taxonomy of network anomaly detection methods according to statistics, clustering, classification, and information theory. PCA is often regarded as one of methods in data mining.

Table 4. Methods of Network Anomaly Detection

Categories	Methods
Statistics	Principal component analysis (PCA), signal processing, mixture model
Clustering	Regular clustering, co-clustering
Classification	Rule-based, support vector machine (SVM), Bayesian network, and neural network
Information theory	Techniques based on entropy, relative entropy, conditional entropy, information gain, or information cost

The *k*-means clustering method can be used to cluster a dataset into a number of clusters. One may directly cluster the training set or alternatively choose to perform feature selection followed by dimensionality reduction and then apply *k*-means clustering on the data with reduced dimensionality [29]. The *k*-means algorithm was chosen to evaluate the performance of an unsupervised learning method for anomaly detection using the KDD Cup 1999 network dataset. The results of the evaluation confirm that a good detection rate can be achieved while maintaining a low false alarm rate [30]. A hybrid IDS with a group of three data mining methods was used for anomaly detection to decrease false alarm rate in the IDS which includes *k*-means, *k*-nearest neighbour (*k*-NN) and the Decision Table Majority method. Another hybrid model that combines *k*-means, *k*-NN, Naive Bayes was presented.

This model uses entropy based feature selection method for attribute selection. It applies *k*-means clustering algorithm for the clustering purpose which is followed by *k*-NN and Naïve Bayes classification algorithms for detecting intrusions. The model shows better performance than only *k*-means or the combination of *k*-means and *k*-NN. The IDS based on the neural network with the back-propagation algorithm requires a very large amount of data and takes time to ensure the results accuracy. Boosted decision tree (DT) approach for intrusion detection system is an ensemble approach and its detection rate is good but has moderate false alarm rate. Because it combines a few decision trees, it becomes complex and needs more time and space [31].

Using a DT for classification gives a good accuracy which in turn can help reduce the training and testing time compared with traditional neural network. The importance of the DT in modelling intrusion detection for the classes R2L and DoS has been proven. For the classes U2R and probe, the rule-based classification is more suitable. However, the DT is more suitable than the rule-based classification in modelling intrusion detection systems based on acceptable levels of the false alarm rate [8]. Some techniques for intrusion detection were investigated and their performance was evaluated based on the benchmark KDD Cup 99 data. DT and SVM were explored as intrusion detection models; a hybrid DT–SVM model and an ensemble approach with DT, SVM and DT–SVM models were designed as base classifiers. Empirical results revealed that DT gives a better or equal accuracy for classes normal, probe, R2L, and U2R. The hybrid DT–SVM approach improves or delivers equal performance for all the classes when compared with a direct SVM approach. The ensemble approach gave the best performance for R2L and probe classes. The ensemble approach gave a 100% accuracy for the probe class, and this suggests that a 100% accuracy might be possible for other classes too if proper base classifiers are chosen [32]. Traditional data mining and machine learning methods have limitations in intrusion detection and prevention because ID/IP systems generate big data with high volume, high velocity, and various data formats, etc.

4. Big Data in Intrusion Detection Systems

4.1. Huge Volume of Data and Data Fusion for Heterogeneous Sources

Traditionally, the range of systems for detecting and preventing cyber-attacks can be grouped as follows: antivirus programs, host IDS/IPS, network IDS/IPS, logging, network device events, file integrity monitoring (FIM) and whitelisting, and security and information event management (SIEM). Although these systems are useful in many ways, they are proving to be largely ineffective against current types of stealthy cyber-attacks. The reasons are: 1) they operate independently from each other; 2) they generate a huge amount of data which is difficult and time consuming to analyse, thus it is easy to miss key cyber-attack events [33]. Big Data analytics (BDA) can sift through a huge amount of data much quicker and heterogeneous systems can be made more efficient and effective [20]. Validating the vast amount of data in content networks is a major challenge because there is a very large number of different types of sources such as blogs, social networking platforms, or news sites with social networking functionalities, and different types of content such as comments, articles, and tweets, etc. Therefore, it is needed to derive simple rules for validating content and leverage content recommendations from other users. The recommending users themselves must be assessed based on the reputation and trust criteria [34].

A challenge in detecting APTs is the massive amount of data to sift for detecting anomalies. The data comes from an ever-increasing number of diverse information sources that have to be audited. Big Data analysis is a suitable approach for APT detection. By using a MapReduce implementation, an APT detection system has the possibility to more efficiently handle highly unstructured data with arbitrary formats that are captured by many types of sensors (e.g., Firewall, IDS, Syslog, NetFlow, and DNS) over long periods of time. In addition, the massive parallel processing mechanism of MapReduce can use much more sophisticated detection algorithms than the traditional SQL-based data systems [9].

A common technique which is used to stop a flood of alerts (big alert data) is called alert correlation. The basic concept of alert correlation is that a system should filter and aggregate multiple alarms into one alarm so that a flood of alarms of the same type does not occur when the same characteristic is causing the same alarm. Data fusion is a technique to aggregate intrusion detection data from many various heterogeneous sources such as system log files, system messages, user profile databases, operator commands, numerous distributed packet sniffers, and Simple Network Management Protocol (SNMP) traps and queries. Situational awareness can be enhanced with data fusion in cyberspace. However, data Fusion has not been widely adopted within the intrusion detection domain. Much research has been conducted for alert correlation with intrusion detection, while few work deals with event fusion or other types of data fusion. More experiment with different data fusion techniques should be conducted, especially in the context of many diverse heterogeneous sources which contain big data [4].

Three main approaches about using Big Data analytics tools for cyber security were discussed. The first method involves making existing systems such as SIEM and data loss prevention (DLP) more intelligent and less noisy so that only the most dangerous cyber-attacks (e.g. APTs) are flagged and isolated. In the second method, the data for the analytics is sourced from internal and external sources (such as online and mobile activities), and the analytics setting is customised (or ad hoc). This means the organisations can set their own search criteria and search for malicious activities can be performed ‘in google-like fashion’ in some Big Data analytics systems. In the third method, the analytics is performed mainly on external data about threats and bad activities. This means that the Big Data analytics system is designed to comb through the Internet (both dark and public) for malicious activities against organisations [20].

4.2. Real-time Stream Data and Big Data Stream Processing

Handling the velocity of big data is not an easy task. First, a system should be able to collect data generated by real-time events streams at a rate of millions of events per seconds. Second, it needs to handle the parallel processing of the data when it is being collected. Third, it should perform event correlation using a Complex Event Processing engine to extract the meaningful information from moving streams. The three steps should happen in a fault tolerant and distributed way. A real-time system should be a low latency system so that computation can be performed very fast with the near real-time response capability [35].

Real-time distributed stream processing models can benefit traffic monitoring applications for cyber security threats detection. Due to the large volume, data should be separated in partition to treat it in parallel. High availability, fault tolerance, and fail recovery are critical in stream processing systems. Stream processing platforms must provide resilience mechanisms against imperfections such as data loss, delays, or out of order samples which are common in data stream. Platforms should minimize communication overhead between distributed processes in data transmission. Real-time monitoring applications require distributed stream processing. The main method to analyse big data in a distribute fashion is the MapReduce technique with Hadoop open-source implementation. Nevertheless, platforms based on this technique are not ideal, even sometimes inappropriate to process real-time streaming applications [36].

Approaches based on traditional solutions like Data Stream Management Systems (DSMS) and Complex Event Processors (CEP) are generally insufficient for the challenges posed by stream processing in a big data context. The analytical tasks required by stream processing are so knowledge-intensive that automated reasoning tasks are also needed. The problem of effective and efficient processing of streams in a big data context is far from being solved even if considering recent breakthroughs in NoSQL databases and parallel processing technologies. Big Data stream processing often poses hard/soft real-time requirements for the identification of significant events because their detection with a too high latency could be completely useless [34]. It was envisioned to build a NIDS that is capable of handling big data network streams by utilizing Big Data tools such as Hadoop and a network monitoring tool called PacketPig. PacketPig is capable of deep packet inspection, deep network analysis, and even full packet capture when using it with Hadoop. The effectiveness of clustering algorithms for analysing packet classification was mainly considered [4].

5. Conclusion

Using a system with multiple types of IDPS technologies helps achieve more accurate and comprehensive performance. Anti-virus products belong to the signature-based detection approach. Signature-based approaches can have high accuracy for existing attacks, but cannot detect new or unknown attacks (zero-day attacks). The anomaly-based approaches can be used to defend against zero-day attacks and can detect most new attacks, but often result in a high false positive rate. Anomaly detectors act as anomaly sensors and each sensor examining one aspect of the host's or user's activities within a network. The triggering of multiple sensors suggests more anomaly behaviour. It is difficult to find out outliers directly from high dimensional datasets; therefore, dimension reduction is often performed by PCA. An ensemble approach often achieves a better performance such as accuracy in modelling intrusion detection and predicting classes of attacks.

Real-time monitoring applications require distributed stream processing. High availability, fault tolerance, and fail recovery are critical in stream processing systems. A key challenge for current IDSs is the limited time window for which the connection state can be maintained. Situational awareness can be enhanced through data fusion that helps deal with diverse heterogeneous data sources and big data. IDS and IPS are sources of big data because they generate a huge amount of data and the data are often heterogeneous. The IDS should be able to recognize the different data formats and different protocols for communication which are the features of big data. Big Data analytics (BDA) can sift through a huge amount of data much quicker and heterogeneous systems can be made more efficient and effective.

References

[1] Choras M., Kozik R., Bruna MPT. Yautsiukhin A, Churchill A, Maciejewska I, & Jomni A. Comprehensive approach to increase cyber security and resilience. In Availability, Reliability and Security (ARES) 2015, 10th International Conference *on* (pp. 686-692). IEEE.

[2] Faisal MA, Aung Z, Williams JR, Sanchez A. Securing advanced metering infrastructure using intrusion detection system with data stream mining. In Pacific-Asia Workshop on Intelligence and Security Informatics 2012 May 29 (pp. 96-111). Springer Berlin Heidelberg.

[3] Scarfone K, Mell P. Guide to intrusion detection and prevention systems (idps). NIST special publication, 2007, 800(2007): 94.

[4] Zuech R, Khoshgoftaar TM, Wald R. Intrusion detection and big heterogeneous data: a survey. Journal of Big Data. 2015, Feb 27; 2(1): 3.

[5] Marchal S, Jiang X, State R, Engel T. A big data architecture for large scale security monitoring. InBig data (BigData Congress), 2014 IEEE international congress on 2014 Jun 27: 56-63. IEEE.

[6] Kizza JM. Guide to computer network security. Springer; 2009.

[7] Kukielka P, Kotulski Z. Analysis of different architectures of neural networks for application in intrusion detection systems. InComputer Science and Information Technology, 2008. IMCSIT 2008. International Multiconference on 2008 Oct 20: 807-811.. IEEE.

[8] Anuar NB, Sallehudin H, Gani A, Zakari O. Identifying false alarm for network intrusion detection system using hybrid data mining and decision tree. Malaysian journal of computer science. 2008; 21(2):101-15.

[9] Cárdenas AA, Manadhata PK, Rajan S. Big data analytics for security intelligence. University of Texas at Dallas@ Cloud Security Alliance. 2013 Sep.

[10] Guillen E, Sánchez J, Paez R. Inefficiency of IDS static anomaly detectors in real-world networks. Future Internet. 2015 May 6; 7(2): 94-109.

[11] Virvilis N, Serrano O. Big Data Analytics for Sophisticated Attack Detection, ISACA Journal, 2014, Volume 3, 1-8.

[12] Vasiliadis G, Antonatos S, Polychronakis M, et al. Gnort: High performance network intrusion detection using graphics processors[C]//Recent Advances in Intrusion Detection. Springer Berlin/Heidelberg, 2008: 116-134.

[13] Raiyn J. A survey of cyber attack detection strategies. International Journal of Security and Its Applications. 2014; 8(1):247-56.

[14] Cisco, Implementing Secure Converged Wide Area Networks (ISCW), Module 6: Cisco IOS Threat Defense Features, 2016.

[15] Zhang L, White G B. An approach to detect executable content for anomaly based network intrusion detection//Parallel and Distributed Processing Symposium, 2007. IPDPS 2007. IEEE International. IEEE, 2007: 1-8.

[16] Singh J, Nene MJ. A survey on machine learning techniques for intrusion detection systems. International Journal of Advanced Research in Computer and Communication Engineering. 2013, Nov; 2(11): 4349-55.

[17] Lu LF, Huang ZH, Ambusaidi MA, Gou KX. A large-scale network data analysis via sparse and low rank reconstruction. Discrete Dynamics in Nature and Society. 2014, May 26; 2014.

[18] Rothman M., Network-based Threat Detection, Technical Report, Securosis, LLC, June 19, 2015, 1-24.

[19] Mai J, Chuah CN, Sridharan A, Ye T, Zang H. Is sampled data sufficient for anomaly detection?. InProceedings of the 6th ACM SIGCOMM conference on Internet measurement 2006 Oct 25, 165-176. ACM.

[20] Oseku-Afful T. The use of Big Data Analytics to protect Critical Information Infrastructures from Cyber-attacks, 2016, 1-64.

[21] Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. ACM computing surveys (CSUR). 2009 Jul 1; 41(3): 15.

[22] Kicanaoglu B. Unsupervised Anomaly Detection in Unstructured Log-Data for Root-Cause-Analysis, Master's Thesis, Computing and Electrical Engineering, Tampere University of Technology, on 4 March 2015.

[23] Manandhar P, Aung Z. Intrusion Detection Based on Outlier Detection Method. ICIDIT ‘2014), April. 2014: 21-2.

[24] Patel A, Taghavi M, Bakhtiyari K, JúNior JC. An intrusion detection and prevention system in cloud computing: A systematic review. Journal of network and computer applications. 2013 Jan 31; 36(1): 25-41.

[25] Tyler G. Information Assurance Tools Report Intrusion Detection Systems. Information Assurance Technology Analysis Center (IATAC), 2009.

[26] Stouten F. Big data analytics attack detection for Critical Information Infrastructure Protection. Thesis, Department of Computer Science, Electrical and Space Engineering, dissertation, Luleå University of Technology, 2016.

[27] Han J, Pei J, Kamber M. Data mining: concepts and techniques. Elsevier; 2011, Jun 9.

[28] De Sanctis M, Bisio I, Araniti G. Data mining algorithms for communication networks control: concepts, survey and guidelines. IEEE Network. 2016, Jan; 30(1):24-9.

[29] Kumar GR, Mangathayaru N, Narsimha G. Intrusion Detection-A Text Mining Based Approach. International Journal of Computer Science and Information Security. 2016, Feb 1; 14: 76.

[30] Nieves JF, Jiao YC. Data clustering for anomaly detection in network intrusion detection. Research Alliance in Math and Science. 2009, Aug 14: 1-2.

[31] Sharma S and Gupta RK, Intrusion Detection System: A Review, International Journal of Security and Its Applications, 2015, Vol. 9, No. 5, 9-76.

[32] Peddabachigari S, Abraham A, Grosan C, Thomas J. Modeling intrusion detection system using hybrid intelligent systems. Journal of network and computer applications. 2007, Jan 31; 30(1):114-32.

[33] Shackleford, D. Using Analytics to Predict Future Attacks and Breaches. [online] SANS Institute, 2016. Available at: http://www.sas.com/content/dam/SAS/en_us/doc/whitepaper2/sans-using-analyticsto-predict-future-attacks-breaches-108130.pdf [Accessed 28 May 2016].

[34] NESSI, Big Data: A New World of Opportunities, NESSI White Paper, December 2012, 1-25.

[35] Bhattacharya D, Mitra M. Analytics on big fast data using real time stream data processing architecture. EMC Corporation; 2013.

[36] Lopez M A, Lobato A G P, Duarte O C M B. A performance comparison of Open-Source stream processing platforms[C]//Global Communications Conference (GLOBECOM), 2016 IEEE. IEEE, 2016: 1-6.