A Framework for Intrusion Detection Based on Workflow Mining

Nkondock Mi Bahanag Nicolas¹, Georges Bell Bitjoka², Emvudu Yves¹

¹Department of Computer Science, Faculty of Science, University of Yaounde I, Yaounde, Cameroon
²Department Telecommunications, National Advanced School of Engineering, University of Yaounde I, Yaounde, Cameroon

Email address:
nikondock.gmail.com (N. Mi. B. Nicolas), georges@bellbitjoka.com (G. B. Bitjoka), yves.emfvidu@minesup.gov.cm (E. Yves)

To cite this article:

Received: August 6, 2019; Accepted: September 6, 2019; Published: September 23, 2019

Abstract: Information systems handle large amount of data within enterprises by offering the possibility to collect, treat, keep and make information available. To achieve this, it is crucial to secure data from intrusion that disturb confidentiality, availability, and integrity of data. This integrity must follow the strategic alignment of the considered enterprise. Unfortunately, the goal of attackers is to affect the resources present in the system. Research in intrusion detection field is still in search of proposals to relevant problems. Many solutions exist supporting machine learning and datamining models. Nevertheless, these solutions based on signature and behavior approaches of intrusion detection, are more interested in data and have not a global view of processes. The aim of this paper is to use workflow mining for a Host-based intrusion detection by monitoring workflow event logs related to resources. With workflow mining, process execution are stored in event logs and the detection of intrusion can be realized by their analysis on the basis of a well-defined security policy. To achieve our goal, step by step, we start by the specification of different concepts manipulated. Afterwards, we provide a model of security policy and a model of intrusion detection that enables us to have a low rate of false alerts. Finally, we implement the solution via a prototype to observe how it can work.

Keywords: Information System Security, Intrusion Detection, False Positive Rate, Workflow Mining

1. Introduction

Nowadays, enterprises use different technologies for the improvement of their business processes, by boosting the quality of service, to be more competitive in the market where needs of users or customers are permanently changing. Nevertheless, like a law of nature, advantages usually generate some problems. In this case, while the quality of service is said to be improved by using powerful technologies, security of data manipulated within an information system appears like a pertinent challenge. We can find several papers on this topic, each of them using a specific approach and presenting advantages and some limits. Behavior and signature approaches for intrusion detection are used, and one of the most challenge faced is the high rate of false alerts when detecting intrusions [1-7]. Several scientists use Data mining and Machine Learning, but the problem of data training to build the model of detection is still present [8-10]. More, existing models concentrate on network traffic data. It explains variance in false alert rate, catalyzed by different new attacks. In this paper, we tackle the issue of false alerts in intrusion detection using workflow mining, particularly to monitor event logs related to resources. Within an information system, all actions that affect confidentiality, integrity, and availability of Information are intrusions. Confidentiality concerns rights and authorizations of users while Integrity is about the reliability of information and of course, every time, data must be accessible in real time by authorized users, this is availability. Everyday information systems are the target of several IT attacks by internal or external attackers. All actions that are not authorized are considered as intrusive and naturally lead to a loss of quality of service. Moreover, wars in the world are managed mainly thanks to IT systems. Interesting solutions for this challenge can be the engine of the development of many countries. We remember for instance the intrusion of the virus called Stuxnet in the Iranian nuclear program. It has affected that
program for two years and imposed considerable financial
damage. Another example of attack is the one realized by
Edward Snowden in the NSA system. His action was
considered as intrusive because he has performed some
malicious task like accessing to sensitive folders without
having permissions. Similar situations are legion in the
world, appear every day, every hour and generate several bad
effects within organizations. These situations show that
intrusion detection is still a big issue in Information System
management. This paper proposes a method to detect
intrusions based on workflow mining. Section 2, based on the
literature review gives details on intrusion detection systems
and explains how the workflow mining works. Part 3
contains the model, and the last one concludes showing
future works.

2. Key Notions

2.1. Intrusion Detection Systems

Intrusion Detection within an information system consists
of the monitoring of different events that occur in the
considered system [11]. Intrusive events are the ones that
contain irregular information, in order not in conformity with
security policy established on the base of the organizational
strategy. Steps of intrusion detection process are firstly
monitoring and analyzing traffic; secondly identifying
abnormal activities; and thirdly assessing severity and raising
the alarm [12]. These steps are executed permanently by the
intrusion detection system in a cyclic way. It exists three
main types of intrusion detection systems [13-15]:

A. HIDS: Host-based Intrusion Detection System -
Controls activities of single equipment, like a computer.
It helps the monitoring of abnormal activities occurring
in a specific machine used in the information system. It
can be a server of an administrator for instance. The
task here reflects activities of different users. The
interest of HIDS is the monitoring of the operating
systems or the applications.

B. NIDS: Network-based Intrusion Detection Systems -
Analyzes traffic existing between computers present in
a network. It can detect irregularity like surcharge of the
system or wrong information present in transferred data.

C. HONEYPOT: It is a computer connected to a net
implementing on purpose a low level of security. The
goal is to distract attackers to protect more sensitive
computers. Moreover, a honeypot is a good way of
discovering new techniques of attack and new tools.

The different intrusion detection systems depending on
their type can catch a large amount of intrusions. Many kind
of attacks can occur in an information system and thus are
considered as intrusive. Some categories of attacks exist:
DoS, DDoS, Scan, U2R, Probe, zero-day [16].

2.1.1. Category of Attacks

Kinds of attacks can occur in an information system and
thus are considered as intrusive. These attacks disturb
confidentiality, availability, and integrity of information
which are core characteristics of a secure environment.

1. Confidentiality: Different users and resources in general
don’t have the same access to data which are not
intended to be especially known publicly. Many levels
of rights is implemented within an information system.
Nevertheless, an attack can make accessible, sensitive
information to an unauthorized person or resource.

2. Availability: Every time, data, resources, the whole
system must be ready and able to produce good results.
3. Integrity: It implies that data are not modified or destroyed by an unauthorized action.
4. Authentication: Different users of the system must provide personal codes, proving their identity. Authentication assumes that it is possible to check provided information to be sure that the user of the system is allowed to have a view on it.
5. Non-repudiation: It is always possible to know the authors of all the tasks realized within the Enterprise.

i. DoS and DDoS attacks

The goal of Denial of Service (DoS) is to attack a system and saturate its resources such that, the considered system becomes unavailable. DDos refers to Distributed Denial of Service. The attacker uses a botnet to achieve his goal. Computers considered as bots are the ones of any users who are not aware that their resources are manipulated to realize an attack. Attacker aims to mobilize spectacular resources to saturate the target. As we can observe in figure 1, he sends a command to the botnet, to flood the target with messages such that it becomes unavailable. It is a very common attack well known by information systems managers because they face it frequently. Webmasters are also the target of this kind of attack when the responsible wants to make their sites face it frequently. It is a very common attack such that it becomes unavailable. It is a very common attack well known by information systems managers because they face it frequently. Webmasters are also the target of this kind of attack when the responsible wants to make their sites unavailable for various reasons.

ii. U2R: User to root

For this type of attack, the attacker has access to a user account on the system and exploits vulnerabilities to illegally gain root access to the system. With root access, the attacker can create several damages in the system and then disturb confidentiality, integrity or availability of data in the system.

iii. R2L: Remote to Local attacks

In the class R2L attack, the attacker sends packets to a machine via the network to illegally get local access. Thus, a remote machine considered as the attacker can send packets via the network and takes advantage of some weaknesses of another machine to gain access to a local account on that machine.

iv. Probe attacks

In a probing attack, the attacker scans a network to amass information suitable (to exploit vulnerabilities).

v. Zero-day attacks

Such attacks are discovered when it appears because the system does not know it.

2.1.2. Signature-based Intrusion Detection

The signature-based detection technique also known as misuse detection techniques allow in detecting and catching intrusions in terms of the characteristics of known attacks or system vulnerabilities [17]. Therefore, any action that conforms to the pattern of a known attack or vulnerability is considered intrusive. This technique refers to techniques that use patterns of known intrusions or weak spots of a system to match and identify intrusions. The sequence of attack actions or activities, the conditions that compromise an information system’s security, as well as the damage left behind by intrusions can be represented by some general pattern matching models.

2.1.3. Anomaly-based Intrusion Detection

Anomaly detection is based on the normal behavior of an actor within an information system, for this end, any action that significantly deviates from the normal behavior is an intrusion. The proposed approach is focuses on a formal and sound description of resources that participate in the execution of identified activities [18]. The anomaly-based intrusion detection techniques allow to detect unusual behavior and thus have the ability to detect symptoms of attacks without specific knowledge of details. They also help in producing information that can, in turn, be used to define signatures for misuse detectors. However, these techniques usually produce a large number of false alarms due to the unpredictable behaviors of users and networks; moreover, they often require extensive training sets of system event records to characterize normal behavior patterns.

2.1.4. Hybrid-based Intrusion Detection

It is the association of misuse and anomaly detection techniques [19]. The goal is to detect unknown intrusion by analyzing and catching abnormal behavior occurring in the system without generating a big amount of false alerts, and at the same time, detect and catch intrusive activities by analyzing and verifying if their signatures are present in the database of the system, containing the list of signatures that represent unauthorized activities. Mainly, hybrid-based intrusion detection technique combines advantages of the precedent techniques and considerably reduce their limits. Research continues in this field and, there is not a perfect hybrid-based intrusion detection model.

2.2. Workflow Mining

To understand workflow mining, it is imperative to know the notion of event logs [20]. An event log is a file that contains the trace of events that occurs in a system. The administrator of the considered system defines the structure of an event log, considering the view he wants to have on certain events that occur in the environment of the system. Workflow mining is a discipline that provides methods to analyze event logs, to extract information from it. Such logs usually contain a big amount of data, and then analysis cannot be made efficiently by the human being. Thus, process mining automatizes analysis of event logs. Core concepts of workflow mining and process mining in general, are process discovery, conformance checking and enhancement of business processes [21-22]. In the present work, workflow mining is used to observe event logs of resources and catch events that violate the security policy, initially defined by the administrator in conformity with the strategic alignment of the enterprise. The activities linked to events that are caught are intrusive. It is conformance checking. Using anomaly, signature or hybrid based intrusion detection techniques, the big amount of interesting models proposed in the literature to detect intrusion, are built with Artificial Intelligence, particularly thanks to classic Machine Learning and data mining models: Neural Networks, SVM, Decision tree, KNN. Workflow Mining can be more useful
for Host Intrusion Detection and thus, can easily help to eradicate attacks, for instance in table 1, we can observe that accuracy of detection is better when workflow mining is used (with the prototype MIBANN). The reason is simple: The interest of network-based Intrusion Detection is on the analysis of packets within the network while Host-based Intrusion is orientated on logs, the starting point of workflow mining [22-23]. Workflow mining theory, for intrusion detection is more interesting than previous Artificial Intelligence models because its models of detection are not built with data examples, but with the security policy. It assures that intrusion founded are events that violates security policy. Then the system generates alarms only for real attacks; it solves the problem of false positive. But, the use of Process Mining at the same time increases the rate of false negative if the rules are not enough to consider the different cases of security violations. This last issue can be addressed by a good definition of important rules for the enterprise because the most important for an enterprise is not to provide the guaranty of 100% of security, but to implement security mechanisms accordingly with the strategy of the enterprise.

3. Formal Framework

This section presents the model used to detect intrusions. Before we describe concepts that we manipulate. It improves some concepts partially defined [24]. For this modeling, we will use abstract data types to represent every concept with a tuple.

3.1. Description of Concepts

3.1.1. The Task

It is an operation that can be realized by a resource of the system. Formally, to define a task we can consider the following tuple:

\[ \text{<Task; TaskName; TaskDesc; Pre; Pos>} \]

Where:

- **Task** is the identifier of the Task,
- **TaskName** the name of Task,
- **TaskDesc** the description of the task,
- **Pre** is the precondition and
- **Pos**, the post condition.

3.1.2. The Quality of Service

It represents the degree of satisfaction of an activity or a process executed within the environment of an information system. If **Cr** represents a list of criteria considered to evaluate the quality of service, **Val** the set of values that can be affected to those criteria and **F** the function defined by

\[ F: \text{Cr} \rightarrow \text{Val}. \]

The quality of service is as follows:

\[ \text{<QoS, Cr, Val, F>} \]

3.1.3. The Event, the Log

An event is a task **Task** that is realized by a specific resource **ResResp** manipulating a set of resources **ResUsed** at a certain date **Date** and associated with a quality of service **QoS**. We automatize it by the following:

\[ \text{<Event; Task; ResResp; ResUsed; Period; QoS>} \]

A Log is simply a set of events:

\[ \text{<Log; Event_Set>} \]

We have built a prototype to validate our model and there, we have proposed a model of events logs in a file (figure 2) such that events can be observable via the interface. (Figure 3)

Figure 2. Event Log Example.
3.1.4. The Resource

Within an enterprise, IT security is implemented in the information system, mainly to protect resources. Here, we represent:

- A log noted Log,
- A type (Human, material) noted Type
- A resource by a description, ResDesc,
- And the list of its sub resources SubRes.

<Res; ResDesc; Log; SubRes; Type>

Remark: A sub resource is also a resource. It is a recursive definition that can be represented by a tree. The real resources are files which are at the leaves of the branches contained in a tree.

3.1.5. The Workflow

It is a way of executing a business process. The goal of monitoring a workflow is to have the trace of documents manipulated in the execution of a business process. Thus, a workflow is a set of the task executed, in a period, manipulating certain resources, to permit the transition of documents between different a beginning work post and an ending work post with quality of service. We define the workflow by the following tuple:
<Wf; TaskSet; ResSet; Period; BeginingPost; EndingPost; QoS>

Figure 4 presents an example of a workflow. Workflow management helps to understand how different documents move in the system during process execution.

3.1.6. The Business Process

A business Process \( Bp \) is a set of activities TaskSet designed to produce in a Period (defined here by the concepts BegininTime and EndingTime), a specific result with some resources Res_Set and reach a specific degree of satisfaction in term of quality of service Qos.

\(<Bp; TaskSet; BegininTime; EndingTime; ResSet; QoS>\)

The analysis of a set of workflows Wf_Set helps to understand the related business process.

Therefore, Bp can be redefined by the following Tuple:

\(<WfSet; QoS>\)

3.2. The Model

First of all we will present the commutative diagram related to our concepts. Secondly, the model of security policy and thirdly, the principal formal model for intrusion detection.

The commutative diagram designed in figure 5 elucidates the interactions between the concepts mentioned above.

3.2.1. The Security Policy

The security policy depends on the enterprise and is a list of rules. Rules Generation follows the previous definition of concepts. For each of them, we generate getters (functions that return values of attributes) and setters (functions that allow modifications of attributes). We can also have certain rules specified by the information system management team like the following:

\( SP=<R_1, R_2, R_3 \ldots R_n> \)

For instance:

\( R1: Task->ResSet \) The list of Resources able to perform a task

\( R2: Res->Period \) The period authorized to use a given resource.

\( R3: Res->ResSet \ast Period \) The set of resources that can modify another one at a predefined period.

We can then define a list of rules that represent security policy.

3.2.2. Detection Model

In this model, intrusions are detected by the difference between descriptive and normative models both represented by sets of logs. A better way to do this is to consider only one file that contains the rules of the security policy. This file permits to parse all event logs, each of them related to a resource. The set of intrusions noted \( ID \) is the set of events Event \( i \) that violates security policy.

Definition: An event \( Ev \) violates the security policy \( SP \) if it violates at least one of the rules of \( SP \).

In this paper, this relation is noted by \( Ev \notin SP \).

Therefore, in an information system, the set ID of intrusions obtained is define as follows:

\[ ID = \{ U Ev_{ij} \in (U Log (Res_j)) / Ev_{ij} \notin SP \} \]

Where \( i \) represents the counter of events and \( j \) is the counter of resources. \((1<i<n \text{ and } 1<j<m)\)

4. Discussion

In this work, we have built a prototype to implement our model of intrusion detection based on process mining. It was deployed on a computer like an HIDS to observe local disk C as we can see in figure 3. That prototype of HIDS has permit to detect intrusions considering a defined security policy. Based on the same security policy, and at the same moment, we have deployed three (03) others IDS where we have described the same rules for the security policy and during 5 days while the computer were used. Especially Snort IDS, Bro IDS and Open IDS where we have classical machine learning and data mining models implemented for intrusion detection.

About our problem of the rate of false alerts, we have seen that, our prototype called MIBANN had a better accuracy in term of...
identification of intrusion. The following table shows accuracy in percentage that we have obtained with the different IDS.

<table>
<thead>
<tr>
<th>IDS</th>
<th>True Positive (Where alert is required)</th>
<th>True Negative</th>
<th>False Positive (False alert)</th>
<th>False Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Snort IDS</td>
<td>91%</td>
<td>82%</td>
<td>17%</td>
<td>11%</td>
</tr>
<tr>
<td>Bro IDS</td>
<td>75%</td>
<td>70%</td>
<td>21%</td>
<td>23%</td>
</tr>
<tr>
<td>Open IDS</td>
<td>88%</td>
<td>55%</td>
<td>13%</td>
<td>12%</td>
</tr>
<tr>
<td>MBANN</td>
<td>98%</td>
<td>97%</td>
<td>2%</td>
<td>5%</td>
</tr>
</tbody>
</table>

After obtaining these results, we have seen that our IDS have a better accuracy. But it can be understood because, it is built such that, the different rules represent an equivalence class of all the rules of the security policy. Thus, intrusions are detected accordingly to the well-defined rules and conformance checking of workflow mining. The results obtained by the three first IDS provide a less accuracy because models implemented have a training phase while the last IDS has its definition more before at the level of the definition of rules.

5. Conclusion and Future Works

From the above mentioned, it appears that intrusion detection is a relevant challenge in information system security. This paper presents a model designed to detect intrusion by workflow mining that permits to analyze event logs presenting events related to resources of the considered system. This approach helps to monitor resources directly and then, detect as intrusive, all actions that violates the security policy built around the rights and permissions defined by the managers of information systems for the manipulation of resources. Moreover, this model provides a solution for the problem of the high rate of false alerts because intrusion do not use training data but the quality of rules that represents the security policy. One of the major challenges to handle here is the management of a large volume of data present in event logs. Another challenge is the manner to find a canonical set of rules for the security policy such that, all the rules can be related to one or many rules present in the set of canonical rules. After experimentation, we have compared our solution to three others solutions and we have found that our prototype deployed on windows system to monitor a local disc C has a better accuracy for detection using workflow mining. The problem of false alerts is managed in another angle and has a better solution. But, this solution implies a good strategy for the construction of security policy. Future works can tackle the challenge of integration of big data techniques to improve intrusion detection within an information system using workflow mining. Moreover, the definition of an automatic model for security policy definition appears like another relevant issue.

References


