World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:10, No:3, 2016

Using Vulnerability to Reduce False Positive Rate in Intrusion Detection Systems

Nadjah Chergui, Narhimene Boustia

Abstract—Intrusion Detection Systems are an essential tool for network security infrastructure. However, IDSs have a serious problem which is the generating of massive number of alerts, most of them are false positive ones which can hide true alerts and make the analyst confused to analyze the right alerts for report the true attacks. The purpose behind this paper is to present a formalism model to perform correlation engine by the reduction of false positive alerts basing on vulnerability contextual information. For that, we propose a formalism model based on non-monotonic $JClassic_{\delta\epsilon}$ description logic augmented with a default (δ) and an exception (ϵ) operator that allows a dynamic inference according to contextual information.

Keywords—Context, exception, default, IDS, Non-monotonic Description Logic $JClassic_{\delta\epsilon}$, vulnerability.

I. Introduction

DEVELOPMENT and performance of information violate mechanisms makes the information security a priority for each infrastructure, which mean an obligation to fix different tools of security that can reduce danger and keep safe the information. Intrusion detection system (IDS) [1] is considered like an important tool whose the objective is gathering and analyzing information in a network in order to detect illegitimate action from inside network as outside. IDS generates alarms to indicate that there are malicious actions. However, intrusion detection system faces a challenge which is the crushing number of alerts generated, whose most of them represent false positive ones. It is estimated that an IDS may generate tens of thousands alerts per day [2]. This massive amount of generated alerts makes the relevant alerts buried under the irrelevant ones, which decrease the efficacy of IDS and make the security operator confused for determine the interesting alerts to report true attacks. In general, false positive alerts are referenced for repeated ones or/and for failed attacks. They due to several reasons, among them (1) the wrong configuration and default setting of software and hardware [3], also (2) the repeated alerts occurred by different IDS installed in the network for the same event,(3) beside that, the IDS can generate alerts for each step of a given multi-step attack. For that, most of works don't focus just to achieve correct identification of attacks but also they aim to eliminate false alerts [4]. Many solution have been proposed in order to eliminate this weakness, among them, alerts correlation approaches [5]-[7] whose the objective is to reduce the overall number of unnecessary alerts using different principles based on information found in alerts, by aggregation and clustering of alerts [8]. Other approaches have interested to increase the number of generated alerts using similarity relationship between features of alerts [9], other solutions have interested to identify logical relationship between alerts for discover causality relation between them [6]. Another type of approaches aims to reduce the number of alerts generated that can drive to the same intrusion in order to construct attack scenario [5]. In fact, using only information found in alerts is not sufficient for reach alerts correlation goals and increase the effectiveness of IDS detection rate, it is interesting to take into account contextual information that can be useful to reduce the number of false positives alerts and enrich the semantic of alerts [10]. For that, the analyst need to define a context under which alerts should be analyzed according to his preferences. In the literature, there are many approaches that discuss this issues [10]-[12]. S. Yahi et al. [13] have used contextual information (like host type, network) represented on description logic (DL) besides a probabilistic graphical model, to allow security operator to analyze alerts in well known context. B. Morin et al. [10] have proposed contextual based model where contextual information is represented on first order logic. A. Sadighian et al. [14] have proposed a flexible context-aware and ontology-based alert correlation framework, their work aims to make automatic the analysis of information resources (type of operational system, applications, user and network) available to the security analyst and preserving maximum flexibility and power of abstraction in the definition and use of concepts. Beside contextual information of the network such topology, localization, cartography,..etc, another type of information like vulnerability can be interesting to identify pertinent alerts and mitigate false positives ones [15], [16]. Vulnerability data is overwhelmed by many research whose the objective is reduce the massive number of false positives alerts using different tools and different techniques [15], [10], [11] . It's important to note that the most of researches that have treated contextual information for improve alerts correlation and reducing false positives are statically, i.e. when the state of the context is changed, the inference stay the same, so the intervention of analyst is needed for update what is changed as information that can make the inference change also. For that, our contribution in this paper aims to propose a dynamic formal model based on vulnerability contextual information to reduce false positives alerts and help the security operator to identify true positive ones depending on his preferences in order to find the relationship between target and attack through vulnerability. We aim to make inference dynamically using $JClassic_{\delta\epsilon}$ non-monotonic description logic with default and exception formalism [17]. $JClassic_{\delta\epsilon}$ is a kind of formalism language

Nadjah Chergui, Narhimene Boustia are with the Computer Science Department, University of, Blida, Algeria (e-mail: nadjah.ch@gmail.com, nboustia@gmail.com).

for knowledge representation and solves specific problems in specific field. This logic is based on algebraic-based semantics which has computation in polynomial time. The motivation behind the uses of non-monotonic description logic $JClassic_{\delta\epsilon}$ is that the classical forms of DLs permit to present concepts using only strict properties, while default knowledge is represented using incidental rules. However most of concepts can't be just defined by the use of strict properties, hence the necessity of introducing the notion of default and exception knowledge. Coupey and Fouquer [18] have developed a new non-monotonic description logic called $AL_{\delta\epsilon}$, that handle with the notion of default and exception in concepts definition. It was elaborated by adding to the description logic AL [19] two connectives: (δ) to represent default facts and (ϵ) exception facts. This language was improved by the addition of connectors from C-classic which permit to augment its expressivity and thus make it usable from a practical point of view. By our contribution, the inference can be dynamically when vulnerability contextual information has changed, the presence of vulnerability is represented by the operator $\delta(default)$ and its lack by exceptional operator $\epsilon(exception)$. The rest of paper is organized as follow, in the Section II we will present $JClassic_{\delta\epsilon}$ formalism, in Section III we will present with detail our proposed model to represent contextual information vulnerability and explain how this can be useful to reduce irrelevant alerts with a case study, in Section IV we discuss the related work concerning alerts correlation and contextual information approaches, at final in Section V we present a conclusion and the future works for reducing irrelevant alerts.

II. Nonmonotonic Description Logic $JClassic_{\delta\epsilon}$

In this section we present non-monotonic description logic $JClassic_{\delta\epsilon}$ developed by N. Boustia et al. [17] it is a description logic augmented with a default(δ) and an exception (ϵ) operators to capture context feature , it is a combination of $AL_{\delta\epsilon}$ a description logic augmented with default and exception [20] and cClassic formalism [21]. $JClassic_{\delta\epsilon}$ is defined using a set of atomic concepts P, a set of atomic roles R, the constants $\top$ (Top) and $\bot$ (Bottom),a set of individuals I called "classical individuals" and the following rules (C and D are concepts,δ (Default) and ϵ (Exception) are unary concepts, $\sqcap$ is a binary conjunction, enables universal quantification on role values, P is a primitive concept, R is a primitive role, u is a real number, n is a integer number and I_i a classical individuals) given in Table I.

Using description logic with default and exception, we can define Bird and Penguin as follow:

$Bird \sqsubseteq Animal \sqcap \delta fly$

This axiom means that all birds are animal and they can fly by default

$Penguin \sqsubseteq Bird \sqcap fly^{\epsilon}$

This axiom means that the penguin is a bird that can't flies by exception, in this case, the concept Penguin inherits from the concept Bird the property Animal but not the property fly since it's an excepted property in the definition of Penguin.

TABLE I
SYNTAX OF $JClassic_{\delta\epsilon}$

C,D $\rightarrow \top$	Universal concept
$\bot$	Bottom concept
P	Atomic concept
$C \sqcap D$	concept conjunction
$\neg P$	negation of primitive concept
$\forall r : C$	C is value restriction on all roles R
Min_u	u is a real number
Max_u	u is a real number
ONE-OF$\{I_1, ..., I_n\}$	concept in extension
R FILLS$\{I_1, ..., I_n\}$	subset of value for R
R AT-LEAST n	cardinality for R (minimum)
R AT-MOST n	cardinality for R (maximum)
δC	default concept
C^{ϵ}	exception to the concept

III. Proposed Contextual Model Based Vulnerability

As well known, IDS generates an overwhelmed number of alerts which have to be analyzed by security operator, however, false positive alerts can hide the relevant ones. For that, the security operator needs to identify some preferences in order to define priority on some objects to protect in the network, and analyze alerts that target this objects. For make that possible, it is interesting to take into account contextual information that can really change security operator preference and drive to change the inference concerning the select of relevant alerts. The vulnerability refers to security imperfection or breaches in software that can be used by the intruder to get access to a system or network [16]. For that, the vulnerability information can be benefit to identify a grand part of false positive alerts. The idea behind this work is that at the first step, the security operator identifies vulnerabilities of objects that he interest to protect (for example a server web), after that, the analyst analyzes all alerts that target this object, and at the same time they refer to attacks that exploit the vulnerabilities identified for this target. Furthermore, this contextual vulnerability information might have changed at any instance since that returns to different reasons (network's update, changing of configuration, figure out the lack of network and fix it,..etc), which make the intervention of security operator an obligation to update the input of inference every time when the contextual information state have changed. for avoid this statical updating, we propose a dynamic model based on $JClassic_{\delta\epsilon}$ non-monotonic description logic with default and exception, it allows to represent the presence of vulnerability using the operator default (δ) and its lack using exception operator (ϵ) that mean that the inference is dynamic. In our approach, we suppose that each alert not selected is not relevant or it is false positive one, so it is sufficient to define only the efficient ones. In this paper, we consider that the source of alerts is the both of IDS types, HIDS and NIDS. Now, we will represent concepts of our proposed model by a DL knowledge base. The DL knowledge base is divided into two components, a TBox and ABox. TBox contains the intentional knowledge in a terminological form and it is built through declarations that describe general properties of concepts. An ABox contains the extensional knowledge, also called assertional knowledge that is specific to the individuals

of domain.

A. Representing The Concepts With Contextual Information (TBOX)

In this section, we are interesting to present our proposed model for the reduction of false positive alerts. we need to define different concepts and rules for illustrate the strength impact of the presence of vulnerability information to eliminate a grand part of unnecessary alerts. In our system, TBox includes the following concepts:

1) Context Concept : we consider all information that has a dependence with the host (machine in the network) like a context. For that, we define the concept context for describe each host in the network with its context. the standard M4D4 [10] has used some attributes such as host address IP and network address IP to identify the host. Beside that we need to identify some particular attributes to describe if this host has a particular vulnerability. The TBox includes the following axioms:
$Address \sqsubseteq \top$
Network$\sqsubseteq \top$
TypeOS$\sqsubseteq \top$
OtherData$\sqsubseteq \top$

We define the concept OS to describe the context or the conditions needed for having a particular vulnerability (see case study):
$Os \sqsubseteq hasName.TypeOS \sqcap hasVersion.Version \sqcap hasName.OtherData$
The concept describe host operational system, it is identified by the type of operational system (TypeOS), its version and a filed other data. The following axiom presents Context concept:
$Context \sqsubseteq HostAd.Address \sqcap HostNk.Network \sqcap HostOs.Os$ Which mean that a context is identified by host IP address, its network IP address and its operational system. HostAd, HostNk and HostOs are binary relations to link respectively
context concept with host IP address.
context concept with host IP network address.
context concept with host operational system.

2) Alert Concept : alert concept is defined to represent alarms generated by IDS, we consider that each alert is represented by attributes defined in IDMEF format (for Intrusion Detection Message Exchange Format), [13] show alert concept represented by description logic formalism; in our system we use the same attributes to describe generated alert and we propose the following representation of alert concept in $JClassic_{\delta\epsilon}$.

Alert $\sqsubseteq \forall messageId.String \sqcap$
= 1 messageId $\sqcap$
hasCreateTime.Time $\sqcap$
= 1 hasCreateTime $\sqcap$
$\forall hasDetectTime.Time \sqcap$
hasDetectTime $AT-LEAST1 \sqcap$
$\forall hasAnalyserTime.Time \sqcap$
hasAnalyserTime
$AT-LEAST1 \sqcap \forall hasAnalyser.Analyser \sqcap$
= 1 hasAnalyser $\sqcap$
$\forall hasSource.Source \sqcap$
$\forall hasTarget.Target \sqcap$
$\forall hasClassification.Classification \sqcap$
= 1 hasClassification $\sqcap$
$\forall hasAssessement.Assessment \sqcap$
hasAssessment $AT-LEAST1 \sqcap$
$\leq hasAdditionalData.AdditionalData$

Such an axiom means that an alert admits a unique identifier which is a string, a unique field detecttime of type time, a unique field createtime of type time, and at most a unique field of type time. Moreover, to an alert, we can associate a (or many) source (resp. target). Besides, an alert has a unique classification, at most a filed assessment and a filed additional data [13].

3) Vulnerability Concept : we consider vulnerability like contextual information depending to target host which is represented by Context concept and the attack executed represented by Attack concept, M4D4 model provide a definition of vulnerability based on the following attributes such as degree of severity which take one value of {low, medium or high}, it is defined also by the necessary access level if it is exploited , type of violated information and its published date, in our system we use vulnerability reference as attribute beside attributes represented in M4D4 model. The TBox includes the following axioms:
Reference $\sqsubseteq \top$
$Severity \sqsubseteq \top$
$Requires \sqsubseteq \top$
$LossType \sqsubseteq \top$
We propose the following definition in $JClassic_{\delta\epsilon}$ of vulnerability:
$Vulnerability \sqsubseteq hasReference.Reference \sqcap$
$VulnS.Severity$ONE-OF{low,medium,high}$\sqcap$
$VulnR.Requires$ONE-OF{remote,local,user}$\sqcap$
$VulnL.LossType$ONE-OF{confidentiality,integrity,
availability, privilege_escalation}$\sqcap VulnP.published\_Date.$
hasReference, VulnS, VulnR, VulnL, VulnP are binary relations to link respectively
Vulnerability concept with Reference concept.
Vulnerability concept with Severity degree.
Vulnerability concept with Requires.
Vulnerability concept with LossType .
Vulnerability concept with published date .

4) Attack Concept : our TBox contains also attack concept; for describing attacks. In [14], the authors define attack by ontology which is defined by its objective. We represent each attack by its classification and by the vulnerability that it exploits. The TBox includes the following axiom:
Classification $\sqsubseteq \top$

We define the following axiom for representing attack concept in $JClassic_{\delta\epsilon}$:
$Attack \sqsubseteq AttackC.Classification \sqcap AttackV.Vulnerability$

Such AttackC and AttackV are binary relationship to link attack concept with classification and vulnerability concept respectively.

B. Representing Relationships with Contextual Information

1) Relationship Between Context Concept and Vulnerability Concept : we define the rule $\delta hasVulnerableTo$ with default to represent a relationship between node (represented by context concept) and vulnerability represented by its reference. A node is with default vulnerable if it satisfies the necessary conditions for the referenced vulnerability. These conditions are represented in context concept like type of operational system OS with its version , application,etc.
$\delta hasVulnerableTo$ is represented as follow:
$\delta hasVulnerableTo.(\exists hasReference.Reference) \sqsubseteq hasVulnCx.Context$
this rule means that the node represented by its context is vulnerable to vulnerability identified by its reference, such hasVulnCx is a binary axiom that link context concept and vulnerability.

2) Representing Rule for Relevant Alerts : Now ,we present the rule $\delta Is\_relevantAlert$, which can determine the relevant alerts depending to the preference of security operator, so the alert is with default relevant if the node target is with default vulnerable to the vulnerability exploited by the attack defined by the classification, the following axiom represent the rule $Is\_relevantAlert$ with default
$\delta Is\_relevantAlert \sqsubseteq \exists ID\_alert.Alert \sqcap RlvAlrtA.Attack \sqcap \delta hasVulnerableTo.(\exists hasReference.Reference)$
this rule means that the alert is relevant with default only if the target machine is with default vulnerable. When vulnerability context state is changed (the machine hasn't any more this vulnerability) we represent it by an exception like following:
$Is\_relevantAlert^{\epsilon} \sqsubseteq \exists ID\_alert.Alert \sqcap RlvAlrtA.Attack(\delta hasVulnerableTo.(\exists hasReference))^{\epsilon}$
here, if we have an exception (the node isn't vulnerable any more), we can't consider this alert as relevant one.

C. ABOX

The ABOX of *K* contains instances about individuals. for show how relevant alert can be deduced with our system, we tack an example of ABOX for the TBOX illustrated in the next section.

D. Case Study

we illustrate an example of ABOX for the correspondent TBOX. For deduce the relevant alert to reduce the false positive ones, our system uses the subsumption$\delta\epsilon$ with default and exception, the algorithm of subsumption is described in [17]. we use the following instance of ABOX in Table II.

Using the instances illustrated in the *ABOX* , the alert (al1) is generated for the attack of classification "NET BIOS DCERPC ISystemActivator bind attempt " that exploit the vulnerability (V1). if we add the following instance to the *ABOX*
$\delta hasVulnerableTo.(\exists hasReference.Reference(CVE2003-0352)) \sqsubseteq hasVulnCx.Context(C1)$
We can deduce that the host which has the type of OS is $MicrosoftWindowsNT$ and that has not the Patch $Q823980$ with the version $\prec 4.0.1381.7224$ is with default vulnerable to the vulnerability referenced $CVE2003-0352$. here, in this case with default context, we can say that this alert is relevant one and the system can deduce by the rule $\delta Is\_relevantAlert(I1) \sqsubseteq \exists ID\_alert.Alert(al1) \sqcap RlvAlrtA.Attack(Att1) \sqcap \delta hasVulnerableTo.(\exists hasReference.Reference(CVE2003-0352))$
which mean that the alert (al1) is with default relevant and it has to be analyzed. In the case where the context is changed, i.e. $\delta hasVulnerableTo.(\exists hasReference.Reference)^{\epsilon}$. In $JClassic_{\delta\epsilon}$, we know that $\delta A^{\epsilon} \equiv A^{\epsilon}$, we obtain:
$Is\_relevantAlert(I2)^{\epsilon} \sqsubseteq \exists ID\_alert.Alert(al1) \sqcap RlvAlrtA.Attack(Att1) \sqcap \delta hasVulnerableTo.(\exists hasReference.Reference(CVE2003-0352))^{\epsilon}$.
and because $Is\_relevantAlert(I2) Is\_relevantAlert(I2)^{\epsilon}$ we can't deduce $Is\_relevantAlert(I2)$, which make the alert is not selected as relevant one. For that, the system considers that only the alerts that are deduced like relevant ones.

IV. Related Work

IDS opens a grand area of research for reach a major goal which is the reduction of generated alerts that have to be analyzed. Several approaches have been proposed with different techniques, many of them interest to reduce the unnecessary alerts as much as possible which is the goal of alert correlation. P.Nig et al., [6] have proposed alert correlation model considering that the most of attacks are composed from a set of elementary steps that are depended between them in the sense that the early steps preparing for the next ones, the authors have represented each alert by its prerequisite actions and its consequences ones, based on predicates of first order logic, the alerts are correlated if a precondition of an alert appear in the consequence of another alert. This approach aims to reduce the number of alerts based only on relationship between them. However, the big gap of this approach is that it is incapable to discover all causal relationships between alerts. Another interesting work focus on correlate alerts that have similar values of features. Valdes et al., [9] have proposed a probabilistic model to group alerts based on similarity relationship, two alerts are grouped if there are a similarity between their corresponding features. This type of approaches is recommended to reduce redundant alerts. Other approaches are interesting to reconstruct attacks to correlate alerts that belong to the same attack scenario, S. benfarhet et al., [5] have proposed a graphical probabilistic model to identify elementary attacks and correlate their alerts, each attack plan is represented by a naive bayesian network. this approach aims to reduce alerts by the the reconstruction of attack and correlate their alerts tack into account information found in alert. F.Cuppen et al. [22] have proposed a basic model to reduce redundant alerts and correlate alert belong to

TABLE II
ABOX

ABOX
Classification(NET BIOS DCERPC ISystemActivator bind attempt);
Reference (CVE2003-0352);
Alert(al1)
OS (O1)$\sqsubseteq$ *hasName(MicrosoftWindowsNT),* $\sqcap\neg$*(hasName.Patch Q823980)* $\sqcap(hasName.rpcss.dl) \sqcap (hasVersion. \prec$ *4.0.1381.7224) ;*
Context (C1) $\sqsubseteq$ *HostAd.Address(192.168.1.1)* $\sqcap$ *HostNk.Network(192.168.0.0)* $\sqcap$ *HostOs.Os(O1)*
Vulnerability (V1) $\sqsubseteq$ *hasReference.Reference(CVE2003-0352)* $\sqcap$ *VulnS.Severity(low)* $\sqcap VulnR.Requires(remote)$
$\sqcap$ *VulnL.LossType (confidentiality)* $\sqcap$*VulnP.published_Date(2003);*
Attack(Att1) $\sqsubseteq$ *AttackC.Classification(NET BIOS DCERPCISystemActivator bind attempt)* $\sqcap$ *AttackV.Vulnerability(V1);*

the same attack using alert management, alert clustering and alert merging function based on first order formalism [23]. In intrusion detection system, the rely on only information found in alert is not sufficient to improve detection rate, hence the necessity to take into account contextual information, this information can make the conclusion or the outcome of correlation engine change, like analyst preferences. K. Tabia et al. [24] have proposed a correlation model based on analyst preference for give to alerts more priority or ignore them. In [25], the authors have developed a new alert correlation approach based on knowledge and preferences of security operator, where the general idea representing only alerts that fit security operator knowledge and preferences. Another approach has been proposed [26] allows the integration of security operator knowledge and preferences to the alert correlation process, these information have been represented using Qualitative Choice Logic (QCL). Other solutions get attention to the necessity of vulnerability assessment to enrich the meaning of alerts and reduce the overwhelmed number of false positives. Victor et al., [3] have proposed an approach to reduce false positive alerts. This work present an operational model for minimization of false positive alarms by the suppression of repeated ones. Gula [15] illustrates the impact of using vulnerability information to improve alerts semantic and to extract a massive number of false positive alerts, in this work the author considers that the vulnerability information can be benefit for alert correlation under the observation that the most of attacks exploit a particular vulnerability to gain access to a system or network. B. Morin et al., [10] have proposed a correlation model based on contextual information (topology and cartography of the network) to reduce false positive alerts, they used an additional data such as vulnerability information to improve the meaning of alerts, in this work, the authors have proposed a set of rules based on first order formalism. Massicotte et al., [11] have pointed out the relation between attack and vulnerability reference , they provide a study based on a combination of Snort signatures , Nessus scripts and Bugtraq vulnerability databases to reduce false positive alerts. A. Sadighian et al. [14] have proposed a flexible context-aware and ontology-based alert correlation framework. Their work aims to reduce the grand number of non relevant alerts and false ones based on contextual information (type of operational system, application installed, user profile) represented by ontology, the correlation engine is implemented based on the OWL description logic (OWL-DL). in this work, the authors used vulnerability information in a sense close to that of our model, but the approach does not allow to make inference dynamically in order to checking every time the presence of vulnerability. These approaches provide different methods to reduce false positive alerts based on security preferences and contextual information, however they provide no special mechanism to modeling state changes dynamically. Our system take advantage of non-monotonic $JClassic_{\delta\epsilon}$ formalism to represent vulnerability contextual information using default and exceptional operator to achieve dynamic inference depending to security operator preferences.

V. Conclusion

In this paper, we present our contribution to eliminate dynamically the overwhelmed number of false positives alerts generated by IDS which make the security operator confused to select the pertinent alerts. we used vulnerability contextual information for improve the quality of alerts, enrich alerts semantic and identify the false positive alerts. Our contribution aims to make the inference dynamically using non-monotonic $JClassic_{\delta\epsilon}$. By our contribution, the analyst can check relevant alerts basing on his preferences and his experience about system vulnerability . In this paper we presented our approach in theoretical point of view. Among the perspectives is to implement this approach using real data, and we plan to complete correlation engine by looking for another issues to improve alerts correlation process.

References

[1] S. Axelsson, "Intrusion detection systems: A survey and taxonomy," Technical report Chalmers University of Technology, Goteborg, Sweden, Tech. Rep., 2000.

[2] T. H. Nguyen, J. Luo, and H. W. Njogu, "Improving the management of ids alerts," *International Journal of Security and Its Applications*, vol. 8, no. 3, pp. 393–406, 2014.

[3] G. J. Victor, M. S. Rao, and V. C. Venkaiah, "Intrusion detection systems-analysis and containment of false positives alerts," *Int. J. Comput. Appl*, vol. 5, no. 8, pp. 27–33, 2010.

[4] G. C. Tjhai, M. Papadaki, S. Furnell, and N. L. Clarke, "Investigating the problem of ids false alarms: An experimental study using snort," in *Proceedings of the IFIP TC 11 23rd International Information Security Conference*. Springer, 2008, pp. 253–267.

[5] S. Benferhat, T. Kenaza, and A. Mokhtari, "A naive bayes approach for detecting coordinated attacks," in *Computer Software and Applications, 2008. COMPSAC'08. 32nd Annual IEEE International*. IEEE, 2008, pp. 704–709.

[6] P. Ning, Y. Cui, and D. S. Reeves, "Constructing attack scenarios through correlation of intrusion alerts," in *Proceedings of the 9th ACM conference on Computer and communications security*. ACM, 2002, pp. 245–254.

[7] H. Debar and A. Wespi, "Aggregation and correlation of intrusion-detection alerts," in *Recent Advances in Intrusion Detection*. Springer, 2001, pp. 85–103.

[8] A. B. Mohamed, N. B. Idris, and B. Shanmugum, "Alert correlation using a novel clustering approach," in *Communication Systems and Network Technologies (CSNT), 2012 International Conference on*. IEEE, 2012, pp. 720–725.
[9] A. Valdes and K. Skinner, "Probabilistic alert correlation," in *Recent advances in intrusion detection*. Springer, 2001, pp. 54–68.
[10] B. Morin, L. Mé, H. Debar, and M. Ducassé, "A logic-based model to support alert correlation in intrusion detection," *Information Fusion*, vol. 10, no. 4, pp. 285–299, 2009.
[11] F. Massicotte, M. Couture, Y. Labiche, and L. Briand, "Context-based intrusion detection using snort, nessus and bugtraq databases." in *PST*, 2005.
[12] A. Sadighian, S. T. Zargar, J. M. Fernandez, and A. Lemay, "Semantic-based context-aware alert fusion for distributed intrusion detection systems," in *Risks and Security of Internet and Systems (CRiSIS), 2013 International Conference on*. IEEE, 2013, pp. 1–6.
[13] S. Yahi, S. Benferhat, and T. Kenaza, "Conflicts handling in cooperative intrusion detection: A description logic approach," in *Tools with Artificial Intelligence (ICTAI), 2010 22nd IEEE International Conference on*, vol. 2. IEEE, 2010, pp. 360–362.
[14] A. Sadighian, J. M. Fernandez, A. Lemay, and S. T. Zargar, "Ontids: A highly flexible context-aware and ontology-based alert correlation framework," in *Foundations and Practice of Security*. Springer, 2014, pp. 161–177.
[15] R. Gula, "Correlating ids alerts with vulnerability information," Tenable Network Security, Revision 4, Tech. Rep., 2011.
[16] J. A. Wang and M. Guo, "Ovm: an ontology for vulnerability management," in *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies*. ACM, 2009, p. 34.
[17] N. Boustia and A. Mokhtari, "A dynamic access control model," *Applied Intelligence*, vol. 36, no. 1, pp. 190–207, 2012.
[18] F. Coupey and C. Fouquer, "Extending conceptual definitions with default knowledge," *Computational Intelligence*, vol. 13, no. 2, pp. 401–456, 1997.
[19] F. Baader, *The description logic handbook: Theory, implementation and applications*. Cambridge university press, 2003.
[20] P. Coupey and C. Fouqueré, "Extending conceptual definitions with default knowledge," *Computational Intelligence*, vol. 13, no. 2, pp. 258–299, 1997.
[21] R. J. Brachman, D. L. McGuinness, P. F. Patel-Schneider, L. A. Resnick, and A. Borgida, "Living with classic: When and how to use a kl-one-like language," *Principles of semantic networks*, vol. 401456, 1991.
[22] F. Cuppens and A. Miege, "Alert correlation in a cooperative intrusion detection framework," in *Security and Privacy, 2002. Proceedings. 2002 IEEE Symposium on*. IEEE, 2002, pp. 202–215.
[23] F. Cuppens, "Managing alerts in a multi-intrusion detection environment," in *acsac*. IEEE, 2001, p. 0022.
[24] K. Tabia, S. Benferhat, P. Leray, and L. Mé, "Alert correlation in intrusion detection: Combining ai-based approaches for exploiting security operators' knowledge and preferences," in *Security and Artificial Intelligence (SecArt)*, 2011, p. NC.
[25] S. Benferhat and K. Sedki, "A preference logic-based approach for alert correlation," *Logics in Security*, p. 20, 2010.
[26] L. Bouzar-Benlabiod, S. Benferhat, and T. Boubana-Tebibel, "Integrating security operator knowledge and preferences to the alert correlation process," in *Machine and Web Intelligence (ICMWI), 2010 International Conference on*, Oct 2010, pp. 416–420.