



Towards a new correlation approach in cooperative intrusion detection

—
Délivrable n°04

Salem BENFERHAT
CRIL

Tayeb KENZA
CRIL et Ecole Militaire Polytechnique d'Alger

Safa YAHI
CRIL



Towards a new correlation approach in cooperative intrusion detection

Salem Benferhat, Tayeb Kenaza, and Safa Yahia
Université Lille-Nord de France
Artois, F-62307 Lens, CRIL, F-62307 Lens
CNRS UMR 8188, F-62307 Lens
{benferhat, kenaza, yahi}@cril.univ-artois.fr

1 Introduction

The security of an information system consists to ensure the confidentiality, the integrity and the disponibility of this system. Among the mecanisms used in order to secure an information system, we can distinguish prevention systems like autentification, access control and firewalls.

However, these mecanisms are not sufficient to protect systems againts attacks. In fact, information systems have several vulnerabilities which correspond to different error of conception, implementation or configuration, which enable attackers to execute their intrusions and bypass prevention mecanims.

Intrusion detection [And80] is a field of security aims at detecting attacks againts the considered system. However, intrusion detection suffers from certain problems. Indeed, certain attacks are not detectec. Besides, some alerts are generated while they do not correspond to any real attack.

On the other hand, cooperative intrusion detection, which implies several intrusions detection systems (IDS) and other analysers such as network mapping systems and vulnerability scaners, offer numerous advantages. Indeed, it gives a global vision, and thus complementary points of view. However, given that the systems used in cooperation are not totally reliable, usually conflict appear. Thus, it is very important to resolve this conflicts in order to exploit this cooperation.

In this paper, we present a new corelation approach in a cooperative intrusion detection context. This approach enables to correlate information generated from different systems, in order to reduce the number of alerts, in particular false positives.

The idea of this approach is to reason, without trivialisation, from information generated from several systems used in the detection process. Besides, given that information used in intrusion detection are very structured (usually expressed in XML), we propose to represent them using description logics, which are very appropriate to represent such information. Moreover, these logics are decidable.

2 A brief refresher on description logics

Description logics (DLs) are a family of knowledge representation languages which enable to represent the knowledge of an application domain in a structured and formally well-understood way.

From a practical point of view, description logics are used in several applications in many domains like semantic web, information retrieval, medicine, natural language, databases, etc.

A DL knowledge base includes two components : a TBOX and ABOX. The TBOX (terminological box) introduces the terminology, i.e., the vocabulary of the domain of the application. As to the ABOX, it contains assertions with respect to instances in terms of this vocabulary.

The vocabulary is a set of concepts, which denote sets of individuals, and roles which refer to binary relations between instances.

The description logic \mathcal{AL} (for *Attributive Language*) has been introduced in [SSS91] as the description logic the least expressive having an interest in practice. The description of concepts in \mathcal{AL} are generated by the following syntactic rule :

$$\begin{array}{l}
 C, D \rightarrow A \mid \quad (\text{atomic concept}) \\
 \top \mid \quad (\text{universal concept}) \\
 \perp \mid \quad (\text{concept bottom}) \\
 \neg A \mid \quad (\text{atomic negation}) \\
 C \sqcap D \mid (\text{intersection}) \\
 \forall R.C \mid (\text{restriction of values}) \\
 \exists R.\top
 \end{array}$$

Other logics, more expressive, can be defined by adding other constructors namely

- Union, denoted by $C \sqcup D$ and interpreted by

$$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}.$$

- the complete existential quantification, denoted by $\exists R.C$ and interpreted by :

$$(\exists R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \exists b, (a, b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}.$$

- number restrictions denoted by $\geq nR$ (restriction at least) and $\leq nR$ (restriction at most) where n is an integer :

$$(\geq nR)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} : |\{b \mid (a, b) \in R^{\mathcal{I}}\}| \geq n\},$$

and

$$(\leq nR)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} : |\{b \mid (a, b) \in R^{\mathcal{I}}\}| \leq n\}.$$

- The negation of general concepts denoted by $\neg C$:

$$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} - C^{\mathcal{I}}.$$

3 Representing information in intrusion detection in DLs

In intrusion detection, information are very structured. Usually, they are represented in XML. For instance, the alert format IDMEF (for Intrusion Detection Message Exchange Format) describes alerts in XML. The same thing holds with respect to OVAL (Open Vulnerability and Assessment Language). However, XML is limited to a syntactic representation. Since, this latter is devoid of semantics, a logical formalisme is needed. In this case, the propositional logic is not very appropriate, since it does not permit to represent the informatin in a structured manner, where the necessity to go beyond this logic in terms of expressivity.

We propose thus to consider a fragment of the logic of first order, namely description logics. Le choice of such logics is motivated by the fact that such logics are convenient to represent structured informations. Also, they are decidable. Moreover, actually, we have actually a good number of description logics which vary in terms of expressivity, and for which the complexity of reasoning are well studied. Besides, several reasonners DL have been proposed such as FACT ++. Most of these reasonners use very sophisticated optimisation technics. In fact, these reasonners are efficient in practise, namely with respect to real life problems.

Information needed to our approach imply firstly the alerts generated by the IDS. For the representation of alerts, we relied on the description of alerts according to IDMEF, which is widely used in intrusion detection. We need also contextual information like the topology and cartography. For this last point, we relied on the model M4D4 [MMDD09]. Lastly, our approach requires the description of vulnerabilities. In this case, we have partially used the model M4D4, while considering other sources of description of vulnerabilities, in particular OVAL.

3.1 Representing IDMEF in description logics

L'IDMEF (for Intrusion Detection Message Exchange Format) is an alert format proposed by the group IDWG (Intrusion Detection Working Group). It is a format implemented in XML. A detailed description of this format is provided by the RFC 4765¹. In particular, an alert in IDMEF admits the following characteristics :

- Identifier
- CreateTime
- DetectTime
- AnalyserTime
- Analyser
- Source
- Target
- Classification

¹ <http://www.ietf.org/rfc/rfc4765.txt>

- Assessment
- AdditionalData

We propose to represent the vocabulary of IDMEF in description logics. For this sake, we use a TBOX which contains definition axiom as well as inclusion axioms.

For instance, the concept of an alert is given by figure 1. 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 source (resp. target) or many. Besides, an alert has a unique classification, at most a filed assessment and a filed additional data.

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

Fig. 1. Concept Alert

Let us consider for instance figure 2 which describes an alert in the IDMEF format. This example is comes from RFC 4765 ² of IDMEF. The description of this example in description logic generates an ABOX which contains the following assertions :

- **Different concepts**
 - Alert(ALR1)
 - Analyser(ANL1)
 - Source(SRC1)
 - Target(TRG1)
 - Classification(CLS1)
 - Node(NOD1)
 - Node(NOD2)
 - Node(NOD3)
 - Reference(RFC1)
 - Address(ADR1)
 - Address(ADR2)
- **The alert**

² <http://www.ietf.org/rfc/rfc4765.txt>

```

<?xml version="1.0" encoding="UTF-8"?>
<idmef:IDMEF-Message xmlns:idmef="http://iana.org/idmef"
  version="1.0">
  <idmef:Alert messageid="abc123456789">
    <idmef:Analyzer analyzerid="hq-dmz-analyzer01">
      <idmef:Node category="dns">
        <idmef:location>Headquarters DMZ Network</idmef:location>
        <idmef:name>analyzer01.example.com</idmef:name>
      </idmef:Node>
    </idmef:Analyzer>
    <idmef:CreateTime ntpstamp="0xbc723b45.0xef449129">
      2000-03-09T10:01:25.93464-05:00
    </idmef:CreateTime>
    <idmef:Source ident="a1b2c3d4">
      <idmef:Node ident="a1b2c3d4-001" category="dns">
        <idmef:name>badguyexample.net</idmef:name>
        <idmef:Address ident="a1b2c3d4-002"
          category="ipv4-net-mask">
          <idmef:address>192.0.2.50</idmef:address>
          <idmef:netmask>255.255.255.255</idmef:netmask>
        </idmef:Address>
      </idmef:Node>
    </idmef:Source>
    <idmef:Target ident="d1c2b3a4">
      <idmef:Node ident="d1c2b3a4-001" category="dns">
        <idmef:Address category="ipv4-addr-hex">
          <idmef:address>0xde796f70</idmef:address>
        </idmef:Address>
      </idmef:Node>
    </idmef:Target>
    <idmef:Classification text="Teardrop detected">
      <idmef:Reference origin="bugtraqid">
        <idmef:name>124</idmef:name>
        <idmef:url>http://www.securityfocus.com/bid/124</idmef:url>
      </idmef:Reference>
    </idmef:Classification>
  </idmef:Alert>

```

Fig. 2. Exemple d'alerte

- messageId(ALR1, "abc123456789")
- hasCreateTime(ALR1, 2000-03-09T10:01:25.93464-05:00)
- hasAnalyzer(ALR1, ANL1)
- hasSource(ALR1, SRC1)
- hasTarget(ALR1, TRG1)
- hasClassification(ALR1, CLS1)
- **The analyser**
 - analyzerId(ANL1, "hq-dmz-analyzer01")
 - hasNode(ANL1, NOD1)
- **The node N1**
 - category(NOD1, "dns")
 - location(NOD1, "Headquarters DMZ Network")
 - name(NOD1, analyzer01.example.com)
- **The source**
 - hasNode(SRC1, NOD2)
 - hasIdent(SRC1, a1b2c3d4-002)
- **The node N2**

- hasIdent(NOD2, "a1b2c3d4-001")
- category(NOD2, "dns")
- name(NOD1, badguy.example.net)
- hasAddress(NOD2, ADR1)
- **The address ADR1**
 - hasIdent(ADR1, "a1b2c3d4-002")
 - category(ADR1, "ipv4-net-mask")
 - address(ADR1, 192.0.2.50)
 - netmask(ADR1, 255.255.255.255)
- **The node N3**
 - hasIdent(NOD3, "d1c2b3a4")
 - category(NOD3, "dns")
 - hasAddress(NOD3, ADR2)
- **The target**
 - hasIdent(TRG1, "d1c2b3a4")
 - hasNode(TRG1, NOD3)
- **The address ADR2**
 - category(ADR2, "ipv4-addr-hex")
 - address(ADR2, 0xde796f70)
- **The classification**
 - text(CLS1, "Teardrop detected")
 - hasReference(CLS1, RFC1)
- **The reference**
 - origin(RFC1, "bugtraqid")
 - name(RFC1, 124)
 - url(RFC, <http://www.securityfocus.com/bid/124>)

3.2 Topology

Describing the topology enables for example to deduce if an IDS is capable to detect an alert. The topology concern the nodes as well as their interconnexions. In the model M4D4, we consider that each network has a unique address. We translate this information in DLs in the following manner :

$$\text{Network} \sqsubseteq \forall \text{netaddress} . \text{String} \sqcap = 1 \text{ netaddress}$$

The node represent any machine connected to the network. A node has an address and belongs to a network.

$$\text{Node} \sqsubseteq \forall \text{nodeaddress} . \text{String} \sqcap = 1 \text{ nodeaddress} \sqcap \\ \forall \text{hasNodeNet} . \text{Network}$$

Gateways are particular nodes whose objective is to connect networks. Clearly, a gateway belongs to more than one network.

$$\text{Gateway} \sqsubseteq \text{Node} \sqcap > 1 \text{ hasNodeNet} \\ \text{Node} \sqcap \neg \text{Gateway} \sqsubseteq = 1 \text{ hasNodeNet}$$

The gateways which are directly accessible by a node are designated by :

$$\text{Node} \sqsubseteq \forall \text{hasNodeGateway}.\text{Gateway} \sqcap \\ \forall \text{hasNodeSystemname}.\text{String} \sqcap = 1 \text{ hasNodeSystemname}$$

Lastly, a node can be characterized by its system name :

$$\text{Node} \sqsubseteq \forall \text{hasNodeSystemName}.\text{String} \sqcap = 1 \text{ hasNodeSystemName}$$

3.3 Cartography in DL

According to the model M4D4, a product is characterized by a unique name, a unique version, a unique type and a unique architecture. In descriptions logics, this corresponds to the following concept product :

$$\text{Software} \sqsubseteq \forall \text{softwareName}.\text{String} \sqcap = 1 \text{ softwareName} \sqcap \\ \forall \text{softwareVersion}.\text{String} \sqcap = 1 \text{ softwareVersion} \sqcap \\ \forall \text{softwareType}.\text{String} \sqcap = 1 \text{ softwareType} \sqcap \\ \forall \text{softwareArchitecture}.\text{String} \sqcap = 1 \text{ softwareArchitecture} \sqcap$$

A node hosts a product :

$$\text{Node} \sqsubseteq \forall \text{hosts}.\text{Software}$$

A process is a product executed by a user.

$$\text{Process} \sqsubseteq \forall \text{hasSoftware}.\text{Software} \sqcap = 1 \text{ hasProduct} \sqcap \\ \forall \text{hasUser}.\text{User} \sqcap = 1 \text{ hasUser}$$

A service is a process which listens on a port :

$$\text{Service} \sqsubseteq \forall \text{hasProcess}.\text{Process} \sqcap = 1 \text{ hasProcess} \sqcap \\ \forall \text{port}.\text{Integer} \sqcap = 1 \text{ port}$$

3.4 Les vulnérabilités en DLs

Generally, a vulnerability is characterized by its severity, the access level required to exploit, its consequences and its publication date.

$$\text{Vulnerability} \sqsubseteq \forall \text{severity}.\{high, medium, low\} \\ \forall \text{requires}.\{remote, local, user\} \\ \forall \text{losstype}.\{confidentiality, integrity, availability, privilege_escalation\} \\ \forall \text{published}.\text{Date}$$

In the model M4D4, we consider that a vulnerability simply affects a list of products. This list is called configuration. Besides, the characteristics of vulnerabilities can be extracted from several sources. For example, the database NVD³ (National Vulnerability Database), the data base OSVDB⁴ Open Source Vulnerability Database as well as the projet OVAL⁵ (Open Vulnerability and Assesment Language) are independant initiative which aim at structuring information related to vulnerabilities.

By analysing these sources, in particular OVAL which is a standard, we have obserbed that the fact that a node is affected by a vulnerability consists generally to check more complexe logical conditions, which imply conjunction, disjunction and negatation. For instance, let us consider the description (according to OVAL) of the vulnerability CVE-2008-0082 given by figure 3. In this description, we clearly see that the condition of the vulnerability is given under the form of a complex logical formula.

Definition id: oval.org.mitre.oval:def:5995		Date: 2008-09-19
Title:	Windows Messenger Information Disclosure Vulnerability	
Description:	An ActiveX control (Messenger.UIAutomation.1) in Windows Messenger 4.7 and 5.1 is marked as safe-for-scripting, which allows remote attackers to control the Messenger application, and "change state," obtain contact information, and establish audio or video connections without notification via unknown vectors.	
Version:	1	Class: vulnerability
Status:	ACCEPTED	Reference(s): CVE-2008-0082
Family:	windows	
Platform(s):	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows Server 2003	Product(s): Windows Messenger 4.7 Windows Messenger 5.1
Definition Synopsis:		
<ul style="list-style-type: none"> • Windows Messenger 4.7 is installed <ul style="list-style-type: none"> ◦ AND the version of msgsc.dll is less than 4.7.0.3002 • OR <ul style="list-style-type: none"> ◦ Windows Messenger 5.1 is installed <ul style="list-style-type: none"> ◦ AND the version of msgsc.dll is less than 5.1.0.715 		

Fig. 3. Vulnérabilité CVE-2008-0082

Therefore, the notion of configuration proposed in M4D4 is not sufficient in order to describe a vulnerability. We propose thus to take advantage of other sources.

So, the fact that a machine is for instance vulnerable with respect to CVE-2008-0082 is described as follows :

$$\begin{aligned}
 & \exists \text{vulnerableTo}.(\exists \text{hasReference}. \text{CVE} - 2008 - 0082) \sqsubseteq \\
 & (\exists \text{host}.(\exists \text{hasName}. \text{WindowsMessenger}4.7) \sqcap \\
 & \exists \text{host}.((\exists \text{hasName}. \text{msgsc.dll}) \sqcap (\exists \text{hasVersion}. \leq 4.7.0.3002))) \sqcup \\
 & (\exists \text{host}.(\exists \text{hasName}. \text{WindowsMessenger}5.1) \sqcap \\
 & \exists \text{host}.((\exists \text{hasName}. \text{msgsc.dll}) \sqcap (\exists \text{hasVersion}. \leq 5.1.0.715)))
 \end{aligned}$$

³ <http://nvd.nist.gov/>

⁴ <http://www.osvdb.org>

⁵ <http://oval.mitre.org/>

4 A Refresher on the Inference from Partially Preordered Belief Bases

4.1 Inference from Totally Preordered Belief Bases

We first recall some popular inference relations from totally preordered belief bases, namely the lexicographic inference [BDC⁺93,Leh95], the inclusion inference [Bre89] and the possibilistic inference [DLP94].

Let (Σ, \leq) be a totally preordered belief base where Σ is a set of formulae and \leq is a total preorder reflecting the priority relation that exists between these formulae. (Σ, \leq) can be viewed as a stratified belief base $\Sigma = S_1 \cup \dots \cup S_m$ such that the formulae in S_i have the same level of priority and have a higher priority than those in S_j with $j > i$.

Definition 1. Let $A, B \in \text{Cons}(\Sigma)$.

- A is **lexicographically preferred** to B , denoted by $A <_{lex} B$, iff $\exists i, 1 \leq i \leq m$ such that $|S_i \cap A| > |S_i \cap B|$ ⁶ and $\forall j, j < i, |S_j \cap B| = |S_j \cap A|$.
- A is preferred to B with respect to the **inclusion preference**, denoted by $A <_{incl} B$, iff $\exists i, 1 \leq i \leq m$ such that $(S_i \cap B) \subset (S_i \cap A)$ and $\forall j, j < i, (S_j \cap B) = (S_j \cap A)$.

Let $\text{Lex}(\Sigma, \leq)$ (resp. $\text{Incl}(\Sigma, \leq)$) denote the set of all the preferred consistent subbases of Σ with respect to $<_{lex}$ (resp. $<_{incl}$), namely $\text{Lex}(\Sigma, \leq) = \text{Min}(\text{Cons}(\Sigma), <_{lex})$ and $\text{Incl}(\Sigma, \leq) = \text{Min}(\text{Cons}(\Sigma), <_{incl})$. Then,

Definition 2. Let ψ be a formula.

- ψ is said to be a **lexicographic consequence** of Σ , denoted by $\Sigma \vdash_{lex} \psi$, iff $\forall B \in \text{Lex}(\Sigma, \leq) : B \models \psi$.
- ψ is said to be an **inclusion consequence** of Σ , denoted by $\Sigma \vdash_{incl} \psi$, iff $\forall B \in \text{Incl}(\Sigma, \leq) : B \models \psi$.

As to the possibilistic inference, it is defined by

Definition 3. A formula ψ is a **possibilistic consequence** of (Σ, \leq) , denoted by $(\Sigma, \leq) \models_{pos} \psi$, iff $(\bigcup_{i=1}^{s-1} S_i) \models \psi$, where s is the smallest index such that $\bigcup_{i=1}^s S_i$ is inconsistent. If $\bigcup_{i=1}^m S_i$ is consistent then $(\Sigma, \leq) \models_{pos} \psi$ iff $(\bigcup_{i=1}^m S_i) \models \psi$.

Let LEX, INCL and POS denote the decision problems respectively associated with \vdash_{lex} , \vdash_{incl} and \vdash_{pos} . Then, it has been shown that LEX is Δ_2^p -complete [CLSS98], INCL is Π_2^p -complete [Neb91] and POS is $\Delta_2^p[O(\log n)]$ -complete [Neb98]. Moreover, the possibilistic inference is more cautious than the inclusion inference which is itself more cautious than the lexicographic inference [BDC⁺93].

⁶ $|A|$ denotes the number of formulae of A .

4.2 Inference Relations from Partially Preordered Belief Bases

A number of inference relations from partially preordered belief bases have been defined by extending the inference relations from totally preordered belief bases recalled in the previous section. Then, the compatible-based lexicographic inference [YBL⁺08] and the partial binary lexicographic inference [YBL⁺08] extend the lexicographic inference. Both the democratic inference [CRS92] and the compatible-based inclusion inference and also by the weak possibilistic inference [BLP04].

Before sketching these inference relations, let us recall the notion of totally preordered belief bases compatible with a given partially preordered belief base (Σ, \preceq) [BLP04]. Intuitively, a totally preordered belief base (Σ, \leq) is said to be compatible with a (Σ, \preceq) iff the total preorder \leq extends or completes the partial preorder \preceq . More formally: 1) $\forall \varphi, \phi \in \Sigma$: if $\varphi \preceq \phi$ then $\varphi \leq \phi$ and 2) $\forall \varphi, \phi \in \Sigma$: if $\varphi \prec \phi$ then $\varphi < \phi$.

We denote by $Comp(\Sigma, \preceq)$ the set of all the totally preordered belief bases compatible with (Σ, \preceq) .

1. Compatible-based Lexicographic Inference: This inference, denoted here by *Cmp-lexicographic inference*, is based on the idea of totally preordered compatible belief bases [YBL⁺08].

Definition 4. Let $B \in Cons(\Sigma)$. B is said to be *Cmp-lexicographically preferred* iff there exists a totally preordered base (Σ, \leq) compatible with (Σ, \preceq) such that B is lexicographically preferred in (Σ, \leq) .

Let $CmpLex(\Sigma, \preceq)$ denote the set of all the *Cmp-lexicographically preferred* consistent subbases: $CmpLex(\Sigma, \preceq) = \bigcup_{(\Sigma, \leq) \in Comp(\Sigma, \preceq)} Lex(\Sigma, \leq)$. Then, a formula ψ is said to be a *Cmp-lexicographic conclusion* of (Σ, \preceq) , denoted by $(\Sigma, \preceq) \Vdash_{lex}^{Cmp} \psi$, iff

$$\forall B \in CmpLex(\Sigma, \preceq), B \models \psi.$$

2. Partial Binary Lexicographic Inference: The idea of this inference which will be denoted by *P-lexicographic inference* is to compare directly two consistent subbases [YBL⁺08]. First, Σ is partitioned as follows $\Sigma = E_1 \cup \dots \cup E_n$ ($n \geq 1$) such that:

- $\forall i, 1 \leq i \leq n$, we have $\forall \varphi, \varphi' \in E_i$: $\varphi \approx \varphi'$,
- $\forall i, 1 \leq i \leq n, \forall j, 1 \leq j \leq n$ with $i \neq j$, we have $\forall \varphi \in E_i, \forall \varphi' \in E_j$: $\varphi \not\approx \varphi'$.

So, each subset E_i represents an equivalence class of Σ with respect to \approx . Then, a preference relation between two equivalence classes E_i and E_j , denoted by \prec_s , is defined by: $E_i \prec_s E_j$ iff $\exists \varphi \in E_i, \exists \varphi' \in E_j$ such that $\varphi \prec \varphi'$. One can easily see that this partition is a generalization of the idea of stratification associated with totally preordered belief bases. Now, the *P-lexicographic preference* between two consistent subbases of a partially preordered belief base (Σ, \preceq) , denoted by \preceq_{lex}^P , is defined as follows:

Definition 5. Let $A, B \in Cons(\Sigma)$. Then, A is said to be *P-lexicographically preferred* to B , denoted by $A \preceq_{lex}^p B$, iff $\forall i, 1 \leq i \leq n$: if $|E_i \cap B| > |E_i \cap A|$ then $\exists j, 1 \leq j \leq n$ such that $|E_j \cap A| > |E_j \cap B|$ and $E_j \prec_s E_i$.

Let $PLex(\Sigma, \preceq) = Min((\Sigma, \preceq), \prec_{lex}^p)$. Then, a formula ψ is a P-lexicographic conclusion of (Σ, \preceq) , denoted by $(\Sigma, \preceq) \Vdash_{lex}^p \psi$, iff

$$\forall B \in PLex(\Sigma, \preceq) : B \models \psi.$$

3. Democratic Inference: The democratic inference [CRS92] is based on the following preference:

Definition 6. Let $A, B \in Cons(\Sigma)$. Then, A is said to be *democratically preferred* to B , denoted by $A \prec_{demo} B$, iff $\forall b \in B/A, \exists a \in A/B$ such that $a \prec b$.

Let $Demo(\Sigma, \preceq) = Min(Cons(\Sigma, \preceq), \prec_{demo})$ denote the set of all the democratically preferred consistent subbases of (Σ, \preceq) . Then, a formula ψ is said to be a democratic conclusion of (Σ, \preceq) , denoted by $(\Sigma, \preceq) \Vdash_{demo} \psi$, iff

$$\forall B \in Demo(\Sigma, \preceq), B \models \psi.$$

4. Compatible-based Inclusion Inference: This inference, denoted here by Cmp-inclusion inference, is also based on the notion of compatible totally pre-ordered belief bases [JB89].

Definition 7. $A \in Cons(\Sigma)$ is said to be a *Cmp-inclusion preferred subbase* iff there exists a compatible (Σ, \preceq) such that $A \in Incl(\Sigma, \preceq)$.

Let $CmpIncl$ denote the set of all the Cmp-inclusion preferred subbases. Then,

$$(\Sigma, \preceq) \Vdash_{incl}^{cmp} \psi \text{ iff } \forall B \in CmpIncl(\Sigma, \preceq), B \models \psi.$$

5. Strong and Weak Possibilistic Inferences: The corresponding preference relations are defined as follows [BLP04].

Definition 8. Let $A, B \in Cons(\Sigma)$. Then,

- A is preferred to B with respect to the strong possibilistic preference, denoted by $A \prec_{pos}^s B$, iff $\exists b \notin B$ such that $\forall a \notin A, b \prec a$.
- A is preferred to B with respect to the weak possibilistic preference, denoted by $A \prec_{pos}^w B$ iff $\forall a \notin A, \exists b \notin B$ such that $b \prec a$.

Let $Pos^s(\Sigma, \preceq)$ and $Pos^w(\Sigma, \preceq)$ denote respectively $Min((\Sigma, \preceq), \prec_{pos}^s)$ and $Min((\Sigma, \preceq), \prec_{pos}^w)$. Then,

- $(\Sigma, \preceq) \Vdash_{pos}^s \psi$ iff $\forall B \in Pos^s(\Sigma, \preceq), B \models \psi$
- $(\Sigma, \preceq) \Vdash_{pos}^w \psi$ iff $\forall B \in Pos^w(\Sigma, \preceq), B \models \psi$.

5 A new correlation approach based on management of inconsistency

The correlation approach we propose is situated in the context of a cooperative intrusion detection. The idea is to have a global vision and several points of view which can be complementary. Moreover, we consider for the attacks which exploit vulnerabilities, a description base of vulnerabilities in order to verify to what extent an attack has been really executed, in the sense that the presence of a vulnerability reinforces the presence of an attack.

Besides, it is well known that analysers used in intrusion detection are not totally reliable. Indeed, for the IDS, the false positive. Consequently, the cooperation in this case can easily generate conflicts. For instance, an IDS generates an alert about an attack which exploits a vulnerability while a mapping system says that the target is not vulnerable. Another example is an IDS generates an alert while another IDS, which is capable to detect such an alert does not.

We propose then to manage these conflicts using a logical approach of inconsistency tolerant reasoning. In particular, we consider the context of partially preordered belief bases. In fact, in intrusion detection, some analysers are comparable between them. However, other analysers are not : comparing an IDS and information generated by a mapping analyser is meaningless.

So, the input of our approach are ;

- the alerts generated by the different IDS,
- the topology and the cartography
- vulnerabilities

The output is a partially preordered belief base. More precisely, a belief base expressed in description logics. Then, by applying an inference relation from partially preordered belief base according to the prudence required by the security operator, we determine the plausibility that a given machine is the target of an alert or not. In the case where we find that this attack is not plausible, the corresponding attack is dropped which reduces the number of alerts to handle.

6 Use case

Let us illustrate the previous approach with a simple use case. We consider an intrusion detection system where are implied two IDS I_A and I_B in addition to a mapping system S . Let us suppose that I_A has the same reliability of I_B . In addition, these two IDS are incomparable with respect to the network mapping S .

Let us suppose the following situation :

- I_A generated an alert about an attack A whose the name is "NETBIOS DCERPC ISystemActivator bind attempt" against the machine M .

$A_1 : hasName(A, "NETBIOS DCERPC ISystemActivator bind attempt")$

$\square attackedBy(M, A)$.

- I_B did not generate an alert with respect to the attack A against the machine M .

$$A_2 : \neg \text{attackedBy}(M, A). \quad (1)$$

- According to S , the machine M hosts Microsoft Windows NT as well as the patch Patch Q823980.

$$A_3 : \text{hasName}(S_1, \text{"MicrosoftWindowsNT"}) \sqcap \text{host}(M, S_1) \\ \sqcap \text{hasName}(S_2, \text{"PatchQ823980"}) \sqcap \text{host}(M, S_2).$$

- Besides, we know that according to Snort documentation alert tcp EXTERNAL_NET any -> HOME_NET 135 ("msg:NETBIOS DCERPC ISystemActivator bind attempt"; flow:to_server,established; content:"|05|"; distance:0; within:1; content:"|0b|"; distance:1; within:1; byte_test:1,&1,0,relative; content:"|A0 01 00 00 00 00 00 00 C0 00 00 00 00 00 46|"; distance:29; within:16; reference:cve,CAN-2003-0352; classtype:attempted-admin; sid:2192; rev:1;)

that A exploits the vulnerability CVE 2003-0352.

$$T_1 : \exists \text{attackedBy}.(\exists \text{hasName}.NETBIOSDCERPCISystemActivatorbindattempt)$$

$$\sqsubseteq \exists \text{vulnerableTo}.(\exists \text{hasReference}.CVE2003 - 0352)$$

- Lastly, according to OVAL, a machine is vulnerable with respect to à CVE 2003-0352 if and only if it checks the following conditions :
 - it hosts Microsoft Windows NT.
 - it does not host Patch Q823980.
 - the version of rpcss.dll on this machine is $< 4.0.1381.7224$.

$$T_2 : \exists \text{vulnerableTo}.(\exists \text{hasReference}.CVE2003 - 0352) \sqsubseteq \\ \exists \text{host}.(\exists \text{hasName}.MicrosoftWindowsNT) \sqcap \\ \neg \exists \text{host}.(\exists \text{hasName}.Patch Q823980) \sqcap \\ \exists \text{host}.((\exists \text{hasName}.rpcss.dll) \sqcap (\exists \text{hasVersion}. \leq 4.0.1381.7224))$$

So, we obtain a belief base partially preordered $((T, A), \preceq)$ such that :

- $T = \{T_1, T_2\}$,
- $A = \{A_1, A_2, A_3\}$.

The partial preorder \preceq on this base is described by figure 4.

First of all, the maximal consistent subbases containing certain information T_1 and T_2 are A et B such that :

- $A = \{T_1, T_2, A_2, A_3\}$,
- $B = \{T_1, T_2, A_1\}$.

One can easily see that $A \prec_{plex} B$ which implies that p-lexicographic inference from (T, A) is equivalent to classical inference from A . Consequently, we deduce $\neg \text{attackedBy}(M, A) : (T, A) \vdash_{plex} \text{attackedBy}(M, A)$ which is intuitively expected. So, the corresponding alert can be removed. This can reduce the number of alerts which can be analysed by the security operator.

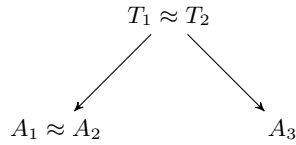


Fig. 4. \preceq over (T, A)

7 Conclusion

In this article, we given a DL representation of information in intrusion detection like IDMEF and M4D4. Moreover, we have shown the pertinence of reasoning in presence of incoherence, in particular, from partially preordered belief bases, in cooperative intrusion detection. We have thus proposed a new logical correlation approach where information are represented in description logics. This approach is parametrized by an inference relation from partially preordered belief bases. Since these relations differ essentially with respect to their prudence, we let such a choice to the security operator.

References

- [And80] J. P. Anderson. Computer security threat monitoring and surveillance. Technical Report 98-17, James P Anderson Co., FortWashington, Pennsylvania, USA, April 1980.
- [BDC⁺93] S. Benferhat, D. Dubois, C. Cayrol, J. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *IJCAI'93*, pages 640-645, 1993.
- [BLP04] S. Benferhat, S. Lagrue, and O. Papini. Reasoning with partially ordered information in a possibilistic framework. *Fuzzy Sets and Systems*, 144:25-41, 2004.
- [Bre89] G. Brewka. Preferred sutheories: an extende logical framework for default reasoning. In *IJCAI'89*, pages 1043-1048, 1989.
- [CLSS98] C. Cayrol, M-C. Lagasquie-Schiex, and T. Schiex. Nonmonotonic reasoning: From complexity to algorithms. *Ann. Math. Artif. Intell*, 22(3-4):207-236, 1998.
- [CRS92] C. Cayrol, V. Royer, and C. Saurel. Management of preferences in assumption-based reasoning. In *IPMU'92*, pages 13-22, 1992.
- [DLP94] D. Dubois, J. Lang, and H. Prade. Possibilistic logic. *Handbook of Logic in Articial Intelligence and Logic Programming*, 3:439-513, 1994.
- [JB89] U. Junker and G. Brewka. Handling partially ordered defaults in TMS. In *IJCAI'89*, pages 1043-1048, 1989.
- [Leh95] D. J. Lehmann. Another perspective on default reasoning. *Ann. Math. Artif. Intell*, 15(1):61-82, 1995.
- [MMDD09] Benjamin Morin, Ludovic Mé, Hervé Debar, and Mireille Ducassé. A logic-based model to support alert correlation in intrusion detection. *Information Fusion*, 10(4):285-299, 2009.

- [Neb91] B. Nebel. Belief revision and default reasoning: Syntax-based approaches. In *KR'91*, pages 417–428, 1991.
- [Neb98] B. Nebel. How hard is it to revise a belief base? In *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, pages 77–145, 1998.
- [SSS91] M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *Artif. Intell.*, 48(1):1–26, 1991.
- [YBL⁺08] S. Yahi, S. Benferhat, S. Lagrue, M. Sérayet, and O. Papini. A lexicographic inference for partially preordered belief bases. In *KR'08*, pages 507–517, 2008.