Case Retrieval Implementation For Intrusion Detection Architecture Based On Multi Agent Systems And Case Based Reasoning Technique

Mohssine EL AJJOURI, Siham BENHADOU, Hicham MEDROMI
Architecture System Team, Laboratory of Research in Engineering (LRI)
Hassan II University, ENSEM
Casablanca, Morocco

Abstract— In the field of security and intrusion detection system, the technique of case based reasoning (CBR) and other methodologies from the field of artificial intelligence can be used to build applications which ensure the accuracy of decisions, this paper describe the architecture of the proposed system, case base, case representation, case retrieval, case reuse and case revise. For the realization of a CBR system for the domain of intrusion detection, we used a JCOLIBRI, java based framework to build our prototype, the case-base for this system is built with an initial set of 5 intrusions cases contained within the case-base.

Index Terms— Intrusion detection; Security; Case based Reasoning; JColibri; Multi Agents systems

IJSER

1 Introduction

Computer networks and systems have become indispensable tools for business operations. They are now deployed in all professional sectors: banking, insurance, medicine and military. Initially isolated from one another, these neworks are now interconnected and the number of access points continues to grow. This phenomenal development is naturally accompanied by an increase in the number of users.

Concerning these networks, users, known or not, are not very mindful of good intentions, they can exploit the vulnerabilities of networks and systems to try to access sensitive information for purpose to read, modify or destroy them, once these neworks have emerged as targets of potential attacks, security become indispensable.

Many tools and means are available, such as firewalls, harware solutions, auditing software or intrusion detection systems (IDS). IDS is one of the most popular answers to the problem posed by : The continuing evolution of the attacks, the appearance of new attacks, and the need to be able to quickly implement new security policies in a network in order to detect and react as quickly as possible to attacks occurring in that network. Intrusion detection can be done manually or automatically, in the manual intrusion detection process, a human analyst checks log files for suspicious signs that may indicate an intrusion.

A system that performs automated intrusion detection is called intrusion detection system (IDS), when an intrusion is discovered by an IDS, typical actions it can take are, for exaple, recording relevant information in a file or database [1], to generate an e-mail alert or a message on a mobile phone.Determine the actual intrusion detected and take cetain actions to stop or prevent it from happening again.Intrusion detection has been studied for more than twenty-five years, the "passive" Intrusion Detection Systems (IDS) have been deployed more and more widely, followed today by "active" Itrusion Prevention Systems (IPS).

In addition, the majority of solutions are based on centralized mechanisms with low capacity to work on dynamic and distributed environments.

To solve these problems, in this context, we propose an architecture of intrusion detection able to work in intelligence with their environment by exploiting the learning agents notion [2].

2 Case Based Reasoning Paradigm

Case-based reasoning is a learning and reasoning technique to solve some of the complexities of implanting an inference engine. From a problem, a case-based system performs a search in its knowledge base, returns a case similar to the problem, extracts the desired solution, adapts it and stores the new case

- *Mohssine EL AJJOURI is currently pursuing Ph.D degree in computer sciences at ENSEM School, Hassan II University, Morocco.E-mail:e.mohssine@gmail.com.*
- *Siham BENHADOU is currently working as Associate Professor in department of computer sciences at ENSEM School, Hassan II University, Morocco. E-mail: siham.benhadou@gmail.com.*
- *Hicham MEDROMI is currently working as Research Director And a Full Professor in department of computer sciences at ENSEM School, Hassan II University, Morocco.Email:hmedromi@yahoo.fr*

generated This makes it possible to reason in fields whose concepts are not totally formalized and where it would be very difficult to formulate a set of rules of inference.
The case-based reasoning is subdivided into several connected steps forming a cycle presented in figure 1.

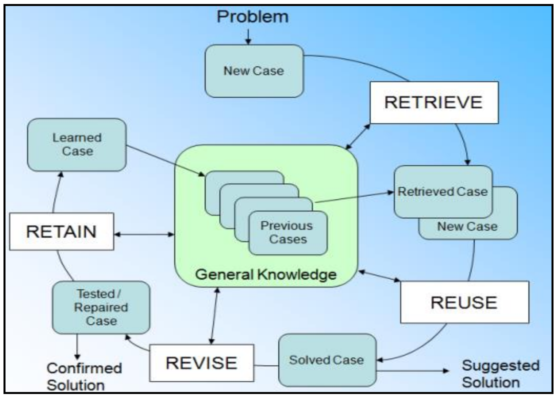


Fig1. A model of Case Based Reasoning System

First, the situation is based on an incomplete set of characteristics describing the problem. This complex step of the process is often performed by a human being entering information into the system, thanks to this brief description of the problem, a retrieve is carried out in a Case-Base to find the previous cases most likely to satisfy the purpose of reasoning, namely to solve the present problem. Return cases depend on both the system case, the cases available in the case base as well as the current reasoning task. The previous solution is extracted from the most similar case for reuse, it becomes a proposed solution, and it's undergoing a process of revision and adaptation in order to become appropriate with the present problem. Adaptation substitutes elements in proposed solution or transforms it for purpose is to make it appropriate in context of new problem. The result of this process generates a new case containing current problem and the confirmed solution which can then be memorized in the case base for future use.

2.1 Base Case

To perform any inference task, a reasoning engine requires knowledge about the target domain [3]. For case-based reasoning, this knowledge forms the base case, this basis is accessible both by the designer of the reasoning system and by the system himself. The user of the system does not need to have access to the entire case base, but must be able to read and understand the content of cases found by the system. A case presents itself in the form of a problem and its solution, of specifications and a plan to satisfy them, A case is a contextual piece of information that describes a successful experience for a particular problem or situation in the past [4].

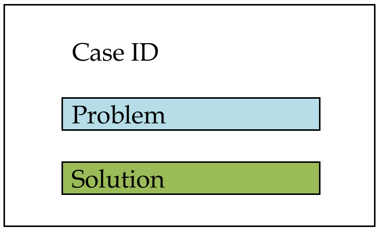


Fig2. Structure of a Case

2.2 Case Representation

An example of a case is presented in Table 1. The case contains two parts: description and solution. The description of the case is a set of characteristics: Protocol, Source IP address, Source port, Destination IP address, Destination port, and Packet contents. In knowledge representation, case-base illustration and vocabulary are standard applications of AI metods and database [5].

TABLE 1
EXAMPLE OF CASE REPRESENTATION

Case Description Attribut
Case ID
Protocol
Source IP Address
Source Port
Destination IP Address
Destination Port
Packets Contents
Case Solution attributs
Output Alert

IJSER

2.3 Similarity Measurements

In CBR applications, essential role is played by notion of similarity, then, purpose of this similarity equation is to find, during a phase of retrieves the more similar case, from a base case. This equation evaluate similarity betwixt two cases is presented below [6].

$$\text{Sim (C, S)} = \sum_{f=1}^{n} W_f \times sim(C_f, S_f)$$

Where:

- C is a current/target case.
- S is a case stored in base case

- n is the number of the attributes/features in each case.
- f is the index for an individual attribute/feature.
- $sim(C_f, S_f)$ is the local similarity function for attribute f in cases C and S.

2.4 Case Retrieval

In our system CBR for intrusion detection, we implemented k-nearest neighbour (K-NN) similarity algorithm, this retrieval method is the most implemented in many successful CBR systems [7].

Our system CBR receive an input case contains information about packets on the networks, this information described by the input case is used by K-NN algorithm to find similar cases from the case base.

The classjcolibri.method.retrieve.NNretrieval.NNScoring method is used in jCOLIBRI framework, this method execute the nearest neighbor on all the attributes, then this method return the most similar cases in the base cases.

3 Architecture of Intrusion Detection Proposed

In this section, architecture of agents is explicated in detail. The agents collaborate between them in to make an efficient distributed intrusion detection framework.

3.1 Motivation of The Proposed Approach

The notion of learning is used to learn normal profiles of users and systems to be secured in current IDS. To detect potentials attacks, normals profiles are compared with the current profiles. we propose in our model to utilize the learning notion to learn abnormal profiles correspond to attacks. it's important to learn new attack patterns for IDS, the aim is to detect new attacks when they reproduce.

One of the main properties of Intelligent Agents is their learning ability. To add this function, the agents use existing patterns attack and attacks that have occurred in the past to identify similarities to a current event suspicious sequence, which does not correspond to a normal sequence or a pattern of attack.

3.2 Multi Agents Paradigm

A multi-agent system is an organized set of agents. it consists of one or more organizations that structure the cohabitation rules and teamwork between agents.

The importance of transition to agent paradigm is compared with importance of using object oriented approach. Agent technology can be effectively applied in different areas of information technology, e.g. computer networks, software development, object-oriented programming, artificial intelligence, human-machine interaction [8].

Artificial intelligence researchers agree on the need for the existence of some features to be able to speak of intelligent agents :

- **Autonomy :** The agent must be able to take initiative and act without intervention from the end user.
- **Ability to communicate and cooperate :** The agent must be able to exchange more or less complex information with other agents, with servers or with humans.
- **Ability to reason, react to their environment :** The agent must be adaptable to its environment. This adaptation must be based on the analysis of the external environment of the agents.
- **Mobility :** Agents must be multi-platform and multi architecture, they must be able to move on the network where they perform tasks without the user having any control over them.

3.3 System Architecture

Proposed approach is IDS distributed and based agent. The agents are observed as proactive, cooperative autonomous, and reflexive entities and also responsible of collecting data and analyzing them. The agents are organized in 3 layers. The functionalities of these agents are briefly illustrated below :

- **Sniffer Agent :** capturing continuously traffic data circulate across a network. Data flow contains information about the packets moving along network to be monitored, traffic flow is captured and split into segments to send through network for another processing.
- **Preprocessor Agent :** after the division of traffic data, the segments produced are pretreated prior to analysis. Once the packet has been observed to have a particular behavior, it is transmitted to the Filtering Agent.
- **Filter Agent :** at this layer, a first analysis be duly executed, so some distrustful events of low complexity will be detected depending on the goals of this agent who search on the rule base, but some events will be skipped, an alarm will be launched the event will be logged in a log file and the administrator will be notified.
- **CBR Agent :** is a core component of our architecture, to achieve target , structure of CBR system is created around the case concept. Case base is a finished set of source

event (S), denoted by BC: {S1, S2, ... Sn}, the source is denoted by S = (PbS, Sol [PbS]), solution of problem source (Sol [PbS]), denoted by Sol [PbS] = {[A1 / V1], [A2 / V2], ... [An / Vn]} where [Ai / Vi] means [attribute / Value]. Agent use attacks occurred in past to identify similarity with current event suspicious sequence. We normalize in table 2 the search result as 0 to 1[9].

TABLE II
SIMILARITY VALUES AND EXPLICATIONS

Similarity value S Between attack A and Attack B	Explications
0	Completely different attack” Attacks A and B have no relationship.
0<S<1	“similar attack” At least the workaround can be similar or the same. So, the countermeasure can be referred the previous cases.
1	Attacks A and B are completely same. So, the countermeasure of attack A can be applied attack B without any modifications.

IJSER

- **Decider agent :** Based on precedent algorithm, the objective for this agent is to verify degree of similarity, if is high then in this case the attack is already known, therefore a classical trigger will be made, it will be archived and the administrator will be informed. Otherwise it is a new attack, for our model it's not known, in that case the agent generator of attack scheme intervenes.
- **Generator Agent :** If the event sequence is identified as an attack, then the generator agent creates a new attack pattern linked to the suspicious event sequence, it will be strored in the base cases, also the security policies will be updated by the administrator , so we learn a new event.

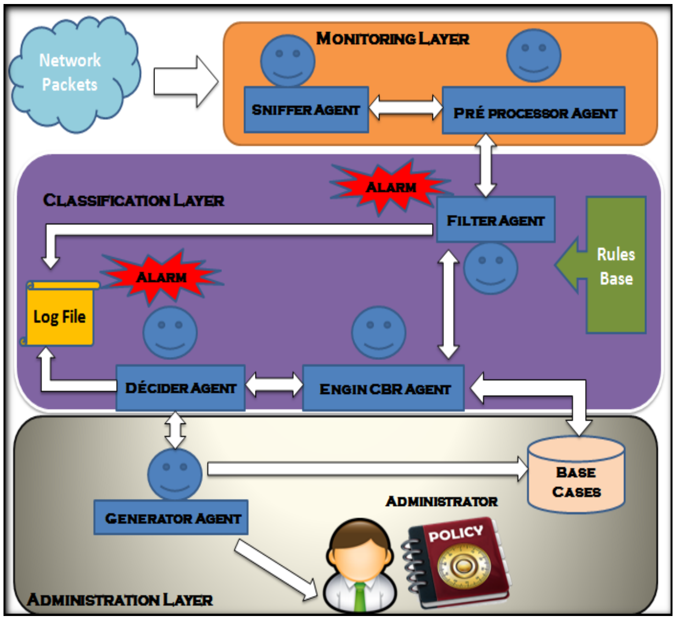


Fig3. Architecture of intrusion detection proposed

4 Implementation Of Our Architecture And Evaluation Results

We use a graphical user interface to test and simplify the process of retrieval in the base case using JAVA and JCOLIBRI framework [10].

4.1 jCOLIBRI Framework

JColibri is a framework developed by the GAIA group [11], that realizes applications in the field of artificial intelligence, their goal is to find new solutions for computer-based education, especially in the field of video games, their research domains are case-based reasoning, knowledge representation, JColibri's project is in the field of case based reasoning, it’s a free and open source java-based CBR framework. It is a comprehensive and efficient tool in developing many types of CBR system applications, varying from textual, structured, knowledge to data intensive systems [12].

We use Eclipse tool as a programming IDE, we create a new java project and we import all JCOLIBRI class in this project.

There are many others tools to build a CBR system, such as CASPIAN, ReCall, CBR-Works, ReMind and MyCBR, jCOLIBRI is the most supported, updated, and has active community of CBR developers [13]. Figure 4 shows the interne architecture of jCOLIBRI.

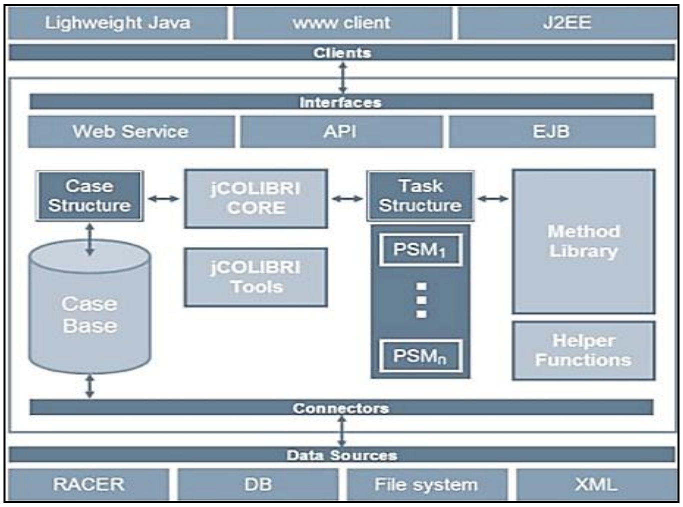
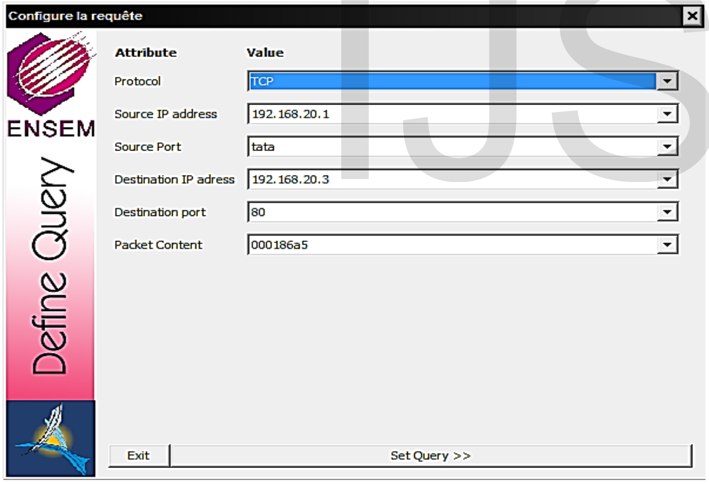


Fig4. Architecture of jCOLIBRI framework

4.1 Our CBR application

In first step, we define our query to the system, the aim is to simulate the reception of different attributes of attacks: Protocol, source ip address, Source port, Destination IP address, Destination port, Packet content as showed in figure 5.

IJSER

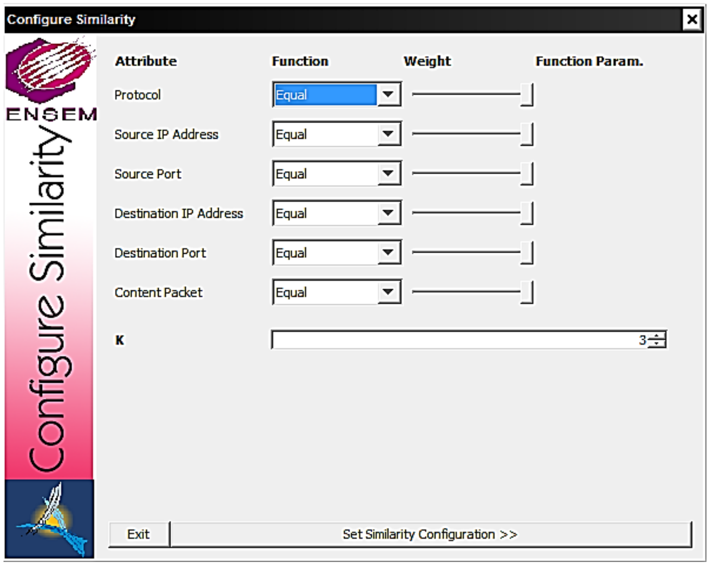


Configure la requête

Attribute	Value
Protocol	TCP
Source IP address	192.168.20.1
Source Port	tata
Destination IP adress	192.168.20.3
Destination port	80
Packet Content	000186a5

Fig5. Step of define query

After, for retrieving the cases most similar, the similarity measure was configured, for this purpose jCOLIBRI has several similarity functions. The K value indicates the number of cases must be retrieved. K Nearest Neighbor (k-NN) algorithm is used to computes the similarity of the query with all cases, then this algorithm return the K most similar cases.



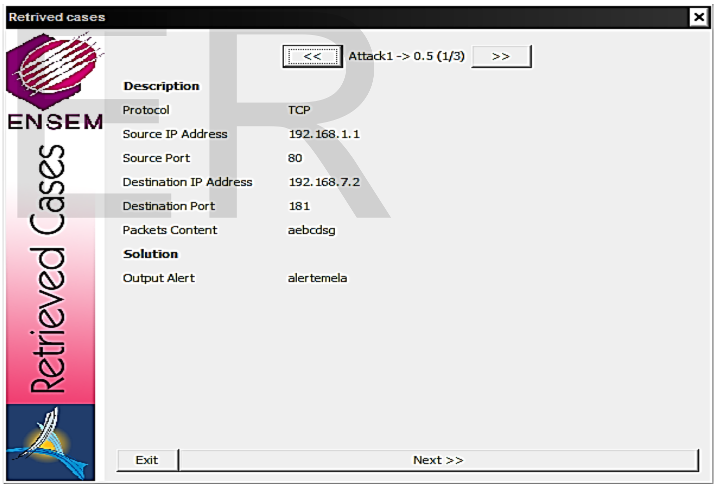
Configure Similarity

Attribute	Function	Weight	Function Param.
Protocol	Equal		
Source IP Address	Equal		
Source Port	Equal		
Destination IP Address	Equal		
Destination Port	Equal		
Content Packet	Equal		

K 3

Fig6. Similarity configuration step

The next step shows the cases returned by K-NN algorithm, in previous window the K=3, then 3 cases most similar was returned as showing in figure 7.



Retrived cases

Attack1 -> 0.5 (1/3)

Description

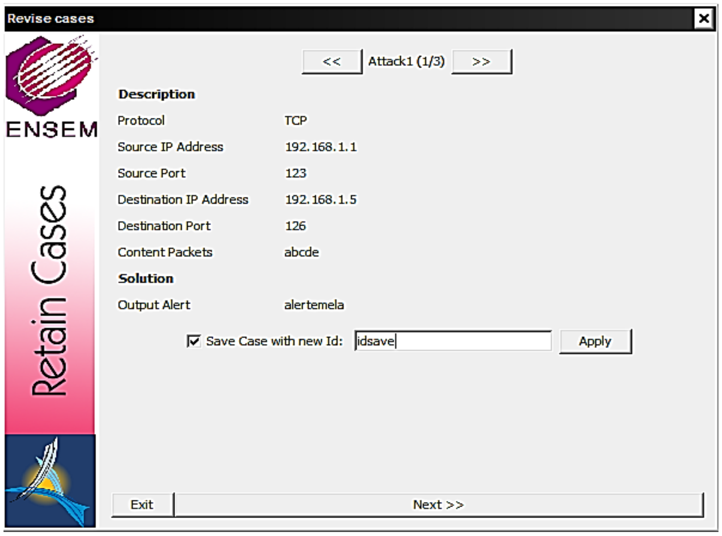
Protocol	TCP
Source IP Address	192.168.1.1
Source Port	80
Destination IP Address	192.168.7.2
Destination Port	181
Packets Content	aebcdsg

Solution

Output Alert	alertemela

Fig7. Step of retrieved cases

Finally, our system would save the new case into the case base , this is used in future query, a new id is assigned to the attack.



Description

Protocol TCP
Source IP Address 192.168.1.1
Source Port 123
Destination IP Address 192.168.1.5
Destination Port 126
Content Packets abcde

Solution

Output Alert alertemela

☑ Save Case with new Id: idsave

Fig8. Step of retain cases

5 Conclusion and Perspectives

In this paper, we have tested case retrieval from the case base for an intrusion detection architecture based on multi-agent systems. In order to do this, we used the CBR technique, which provides our system with an intelligent reasoning mechanism based on past cases. This architecture formed by MAS and present interoperation between agents also distribution of detection activities. For perspective work, we will test our architecture on a real hardware platform consisting of two machines : attacker and victim for purpose of testing various features and ensure that the objective of our model is achieved.

Acknowledgment

I would like to thank my advisors Pr. Siham BENHADOU and Pr.Hicham MEDROMI , for their invaluable guidance and many useful suggestions during my work on this paper.

References

[1] N.Dagorn.(2006). “Détection et prévention d’intrusion : présentation et limites,” Université de Nancy1, [Rapport de recherche] <inria-00084202>.

[2] M, El ajjouri,S, Benhadou; H, Medromi. (January 2016). “New Collaborative Intrusion Detection Architecture Based on Multi Agent Systems,” Journal of communication and computer, vol. 13 , pp. 1-10.

[3] E, Buist. (Février 2004) “Les elements fondamentaux du raisonnement à base de cas,” Université de Montréal.

[4] W,Zanoramy; A, Zakaria. (2015) “Application of case based reasoning in IT Security Incident Response,” Journal of Computing, Communications & Instrumentation Engg. (IJCCIE) Vol. 2, Issue 1 ISSN 2349-1469 EISSN 2349-1477.

[5] B, Ralph; A, Klaus-Dieter.Minor M.Reichle ,Meike; B, Kerstin. (February 2009) “Case-Based Reasoning - Introduction and Recent Developments,” Künstliche Intelligenz: Special Issue on Case-Based Reasoning 23(1), 5-11.

[6] S, Begum; S, Barua; M. U. Ahmed. Jul.(2014) “Physiological Sensor Signals Classification for Healthcare using Sensor Data Fusion and Case-Based Reasoning,” Sensors (Special Issue on Sensors Data Fusion for Healthcare), vol. 7, Jul.

[7] W, Zanoramy Zakaria; M, Laiha Mat Kiah. (2014) “Implementing a CBR Recommender for Honeypot Configuration using jCOLIBRI ,” The 3rd International Conference on Computer Science and Computational Mathematics, (ICCSCM).

[8] N, Kussul; A, Shelestov; A, Sidorenko; V, Pasechnik; S, Skakun; Y, Veremeyenko; N, Levchenko “Multi-Agent Security System Based On Neural Network Model Of User's Behavior,” International Journal Information Theories & Applications" Vol.10.

[9] M, El ajjouri; S. Benhadou; H, Medromi. (2016) “New Model Of Intrusion Detection Based On Multi Agent Systems And CBR Paradigm,” 4th IEEE International Colloquium on Information Science and Technology (CiSt), p133-138.

[10] https://sourceforge.net/projects/jcolibri-cbr/

[11] http://gaia.fdi.ucm.es/

[12] Recio, Garcia; J, Diaz Agudo; B, Gonzalez Calero. (2009) “Boosting the Performance of CBR Applications with jCOLIBRI,” 21st IEEE International Conference on Tools with Artificial Intelligence, 276 – 283.

[13] N, Dendani-Hadiby; M, Tarek Khadir. (May 2016) “Comparative Analysis of Case Retrieval Implementation for Knowledge Intensive CBR Application,” Journal of Advances in Information Technology Vol. 7, No. 2.

[14] M, DAMME ; R. TAVENARD ; S, VENZIN. jCOLIBRI. (2006) “Un atelier générique pour le raisonnement à partir de cas,” 10 mars.

IJSER