Case-Based Cybersecurity Incident Resolution

Marcelo Colome, Raul Ceretta Nunes, and Luis Alvaro de Lima Silva
Graduate Program in Computer Science - PPGCC
Applied Computing Department, Federal University of Santa Maria
Av. Roraima, Campus UFSM, Camobi, Santa Maria, RS, Brazil
{marcelocolome, ceretta, luisalvaro}@inf.ufsm.br

***Abstract* - Intelligent computing techniques have a paramount importance to the treatment of cybersecurity incidents. In such Artificial Intelligence (AI) context, while most of the algorithms explored in the cybersecurity domain aim to present solutions to intrusion detection problems, these algorithms seldom approach the correction procedures that are explored in the resolution of cybersecurity incident problems that already took place. In practice, knowledge regarding cybersecurity resolution data and procedures is being under-used in the development of intelligent cybersecurity systems, sometimes even lost and not used at all. In this context, this work proposes to integrate Case-Based Reasoning techniques and IODEF standard in order to retain concrete problem-solving experiences of cybersecurity incident resolution to be reused in the resolution of new incidents. Experimental results so far obtained with a Case-based Cybersecurity Incident Resolution System (CbCSecIRS) implemented show that information security knowledge can be retained in a reusable memory, so improving the resolution of new cybersecurity problems.**

Keywords— Cybersecurity incidents; case-base reasoning; information security.

I. INTRODUCTION

Information security issues have a critical impact on business mainly because the treatment of security incidents is highly expensive and time consuming to organizations. According to the ISO/IEC 27035 [1], processes of information security management should be grounded on approaches to the capture, structuring, and dissemination of security knowledge in each part of the organization. Fundamentally, security knowledge is expressed as various kinds of lessons learned constructed and refined over the time by cybersecurity experts about how to identify and treat cybersecurity incidents. As investigated here, this knowledge must be retained and reused systematically so that cybersecurity problems could be effectively approached. When such knowledge-based solutions are reused, for instance, the cost of reapplying them (instead of reconstructing them from scratch) each time a cybersecurity incident occurs can be reduced significantly. In crisis situations due to the occurrence of cybersecurity incidents, the collection and representation of incident resolution (CSecIR) procedures is fundamental to organizations since they can be revisited by security analysts as to promptly and comprehensively approach the treatment of cybersecurity issues.

The use of central systems in the collection, correlation, and analysis of data related to security incidents is a common practice in many countries, where Computer Emergency Response Teams (CERT) commonly detect and report thousands of cybersecurity incidents a year. Each time an incident is reported by the CERT, it should be analyzed and solved by the Computer Security Incident Response Team (CSIRT) which is in charge of managing the computer network where the incident took place. To help the sharing of security information exchanged between CSIRTs or other operational security teams, the Internet Engineering Task Force (IETF) has proposed the Incident Object Description Exchange Format (IODEF) [2], which is a format directed to the broad representation of computer security information. Moreover, an IODEF extension aiming to facilitate the representation and exchange of enriched cybersecurity information was also proposed [3, 4]. Despite these efforts, it is still challenging to reuse security solutions [5, 6], especially those derived from concrete experiences of cybersecurity incident problem-solving.

In Artificial Intelligence (AI), while machine learning algorithms explored in the cybersecurity domain are aimed at presenting reliable intrusion detection solutions, these algorithms seldom approach the representation and reasoning with cybersecurity incident resolution procedural knowledge. In practice, such cybersecurity knowledge is being under-used in the development of intelligent cybersecurity systems, decreasing the effectiveness of Cybersecurity Incident Resolution Systems (CSecIRS). With the help of the Case-Based Reasoning (CBR) techniques [7], this paper approaches the collection and representation of this knowledge in the form of cases. Importantly, such cybersecurity incident resolution cases can be shared and reused as part of fundamental case-based knowledge management tasks [8, 9]. In this context, this paper shows how to build Case-based Cybersecurity Incident Resolution Systems (CbCSecIRS) based on information security attributes detailed according to the IODEF standard, including its cybersecurity extension. Instead of acting as a single cybersecurity solution, the overall idea of following the IODEF pattern is to permit to integrate the CbCSecIRS representation and reasoning capabilities from both intrusion detection systems and cybersecurity incident resolution systems.

The paper is structured as follows: Section II describes how cybersecurity incidents are approached and Section III presents related works where CBR techniques are explored in the cybersecurity domain. While Section IV presents our CbCSecIRS proposal, Section V describes experiments and results so far developed in our project. Finally, conclusions are presented in Section VI.

DOI reference number: 10.18293/SEKE2019-204

II. The Resolution and Representation of Cybersecurity Incidents

Knowledge regarding the resolution of cybersecurity incidents is a crucial asset to organizations. To be competitive, large amount of resources are being invested by security companies in order to not lose their valuable cybersecurity incident resolution experiences. By maintaining such lessons learned in a reusable memory, security analysts have the means of avoiding the costly reconstruction of "new" security solutions each time a cybersecurity incident problem occurs. Although large amount of data about incidents is being collected and explored by security companies via different AI approaches, the ISO/IEC 27035 standard [1] states that the processes of cybersecurity incident treatment can be organized in different activities: *i*) *Plan and prepare*: aim to develop incident treatment plans, check-lists of tasks to be executed when such cybersecurity threads occur, and communication plans aiming to record information about how entities involved should be prepare to communicate in the occurrence of security calamities; *ii*) *Detect and report*: as recommended in [10], multiples forms of reporting the cybersecurity incidents should be explored. In addition to the manual reporting, cybersecurity incidents can be reported automatically by security services or other entities as CERTs; *iii*) *Evaluate and decide*: the concrete occurrence of the cybersecurity incident should be evaluated, as well as the magnitude and consequences of such incident. Once this evaluation is developed, the origin of the cybersecurity incident can be traced properly; *iv*) *Respond*: involves the incident treatment actions that are properly planned in advance. Based on such treatment plans, recommended problem-solving steps aimed to deal with the cybersecurity incidents are executed. It means that appropriate resolution actions should be taken as to recover from the cybersecurity incident, in addition to incident documentation and communication to stakeholders; *v*) *Record*: the recording of the lessons learned should start as soon as the cybersecurity incident is closed. In doing so, this recording aims to assess whether the solution designed by the CSIRT was successful. An important task here is to document the cybersecurity incident, including not only its categorization but also its procedures of treatment.

In this paper, the techniques proposed are concerned with the outputs of the *detect and report* activities, retrieving past cybersecurity incident solutions that are relevant to the development of *evaluate and decide* activities. Then, cybersecurity incident resolution plans retrieved are used in *respond* activities, permitting to construct new plans to be explored in the *record* activities. So, a typical problem in such cybersecurity incident resolution scenario is the maintenance of lessons learned. We highlight such lessons are not only captured by the recording of factual information of cybersecurity incidents. In practice, alternative machine learning techniques can be successfully explored in the learning of how to automatically detect cybersecurity threads from such factual data. What we highlight in this work is that these lessons are also formed by the treatment procedures used by security analysts in the resolution of cybersecurity incident problems. So, this concrete experience-based knowledge ought to be collected and stored so that it can be shared among different security systems, in addition of being queried and reused as to better solve new cybersecurity incidents.

In the processes of cybersecurity incident treatment, the IODEF standard defines a *data format* directed to the representation and exchanging of information about cybersecurity incidents [3, 4]. The IODEF data model includes data about hosts, networks and services; attack methodologies and forensic pieces of evidence; incident impact; and approach to document the cybersecurity investigation and treatment workflow. This standard also provides a framework to share the incident information that is usually exchanged by CSIRTs as to facilitate the machine-processing of such information. In essence, the IODEF data format is organized in set of data classes, derived from a basic class *Document* that contains one or more *Incident* class. Each aggregated Incident class describes in its derived classes commonly exchanged information when reporting or sharing derived analysis from security incidents. The cybersecurity incident IODEF extension [3, 4] increased Incident class representation capabilities. Despite the large number of resources provided by the IODEF, as it was developed to be adaptable to the different organizational needs, the classes that are required to represent a cybersecurity problem are of particular importance as this paper shows how a CbCSecIRS can explore them in the representation of concrete experiences of cybersecurity resolution problems (details in the section IV).

III. Case-Based Reasoning in the Cybersecurity Domain

In AI, Case-Based Reasoning [7] relies on a lazy-learning approach to machine learning which focuses the resolution of new problems by reusing solutions recorded in past problem-solving experiences represented as "cases". Given a new problem to be solved as a query in such CBR systems, the key problem-solving steps are 1) the retrieval of similar cases from a case base, 2) the reuse of solutions recorded in the most similar cases retrieved, 3) the revision of such retrieved solutions as to deal with possible differences between past and new case situations and 4) the retention of new case-based problem-solving experiences in the case base as a way of learning how to solve new problems. Relevant works with CBR in cybersecurity research context follow.

In [11], a CBR system explores the organization of attack cases, where a hierarchical structure containing attributes from possible attack situations is used in the representation of such problem cases. To detail the solutions of such cases, the textual description of countermeasures and the user satisfaction degree for solution proposals are used. Although this work presents a relevant solution for this cybersecurity knowledge management problem, it only approach a limited set of response types to incidents.

With the use of CBR, [12] details a RFM (*Recency*, *Frequency*, *Monetary*) technique aimed at reducing false alerts. Considering how recent the security event occurred, its frequency and attributes values, this approach relies on the statistical analysis of log files to detect anomalies. Then CBR is applied on the identification of attack patterns that are similar to past ones. This work is also focused on the incident detection and determination of security event responses, where such responses are expressed as commands to computer security

services. However, this work does not explore the collection and representation of response plans to the treatment of cybersecurity incidents.

In [13], ontologies are integrated to CBR techniques in order to construct a decision-making and response system to the treatment of cybersecurity incidents. In particular, the ontology model is used in the standardized representation of such incidents, resulting on a hierarchical organization of attack types. While this work does not follow cybersecurity representation standards, the collection of automated attack information and manual attack information are the inputs of the resulting CBR system.

In [14], a CBR system to support the construction of cybersecurity incident responses is described. Using information from past attack cases, this system classifies new attacks to better maintain a secure network. While each attack is represented by a sequence of events, each response is represented by a partially ordered set of resolution actions. These attacks are compared with past attack cases stored in a case base, allowing the reuse of response plans recorded as a solution to the new attack situation. Although this work considers the determination of responses to cybersecurity incidents, it is mostly focused on the incident detection through CBR.

From such works, it is possible to state that the exploration of CBR techniques in the cybersecurity domain is limited and the benefits due to the integration of such AI technique with cybersecurity data standards are still open to investigation. Relying on the proposal of a CbCSecIRS proposal, this paper aims to further approach this gap.

IV. A Case-Based Reasoning Model for Cybersecurity Incident Recording and Resolution

The recording and reasoning with expert knowledge regarding to the resolution of cybersecurity incidents is crucial to the effective treatment of new incident problems. In our Case-based Cybersecurity Incident Resolution System (CbCSecIRS) this knowledge is approached as concrete experiences of problem-solving modeled as *cases*. Once such cases stored in a *case base* are available for similarity-based computations, detailed experience-based answers to the resolution of cybersecurity incidents can be better reused by security analysts. In practice, concrete cybersecurity incidents are recorded in a shared memory, allowing security teams to maintain reusable security treatment knowledge.

To allow cybersecurity incident cases (represented as problem-solution pairs) to be reused, the first modeling task is to represent the problem (incident) according to the IODEF standard. In this way, such incident representation is in conformity with other security proposals directed to the improvement of the operational capabilities of CSIRT teams [3, 4]. Once the incident representation complies with IODEF standard, the CbCSecIRS can communicate with other security systems to allow the acquisition/exchange of cybersecurity incident cases (i.e. problem part of such cases). In addition, security logs received along with incident descriptions can also be examined by security analysts as part of the case acquisition and representation tasks.

The case-based process of cybersecurity incident treatment starts when incidents represented in IODEF are captured by the security analysts. Using the CbCSecIRS, concrete occurrences of new cybersecurity incidents are taken as queries. Once retrieved cases (similar to the current incident situation) are available for examination, the incident treatment plans recorded in the cases retrieved can be re-executed. When such proposed solutions prove to be effective in the resolution of the current problem, such new experience of problem-solving can be recorded in the case base as part of a continuous improvement of the case knowledge which is maintained by the system. If there isn't a good solution and a new resolution is planned and executed, it also can be recorded in the case base. Such recordings allow the CbCSecIRS to dynamically learn new cases as to augment its capabilities of solving cybersecurity incidents.

A. The Case Base Modeling

In the modeling of a case, an incident (problem) is represented by a set of attributes and values along with the incident resolution (treatment plan) expressed by a set of actions. Each incident presents particular behaviors and requires particular attributes to be recognized. Thus, a cybersecurity incident in the CbCSecIRS is modeled by incident type, where types considered in our project are listed in Table I.

TABLE I. Incident Types Modeled in The Case Base

Type	Description
Bot	An organization asset starts to be part of a malware infected computer network. The computers of this network are controlled by hackers (*botmasters*)
DoS	Deny of service attack. An inundation attack against a target (host or service) to turn it unavailable. This cybersecurity incident can be centralized or distributed (DDoS)
Proxy	A *proxy* server is infected in order to make anonymous the hackers that are using it. So, such anonymous hackers use the proxy server to make other attacks
MaliciousURL	It is a computer storing malicious files which are accessible by a URL
Copyright	A *host* shares or received protected material by copyright
Spam	Unsolicited message sent from a host to other users
Scan	A *host* scans other host ports in order to find vulnerabilities that may allow an attack
LoginAttempt	Login attempts by brute force in a service account. The overall aim is to obtain an un-authorized access on the system
Phishing	It is an attempt of deceive a legal user using a fake web page with is similar to a correct one
Defacement	Content modification of legal web site without authorization

The attribute selection by incident type derived from the incident characterization detailed in [11-14]. After the identification of such set of attributes from literature, its consistency was checked against the cybersecurity incident reported by the

Brazilian academic network CSIRT. While there are attributes that are common to different types of cybersecurity incidents, others are specific to one type. As a result, eight common attributes and twelve specific attributes were detected and selected to model the incidents in a case. Despite our selection, we highlight the expert can include others when necessary.

To represent the incident case, the modeled cybersecurity incidents were mapped to IODEF format. Figure 1 illustrates how the standard IODEF classes were adapted to support our case model. The *Incident* class derives from *IODEF-Document* class. It is mandatory in IODEF format. The *IODEF-Document* class contains the attributes `version` and `lang` that according to RFC4646 [15] must ever be filled. The *Incident* class expresses a standardized description of commonly shared incident attributes. It specifies the time the incident is reported (`DetectTime`) along with a textual description of the incident (`Description`). The `purpose` attribute is mandatory and it is used to express the reason by which the IODEF document was created (traceback, mitigation, reporting, other). The *Flow*, *System*, *Node*, *Address*, and *OperatingSystem* classes describe environment features involved in the cybersecurity incident. The *Method* class describes the method used in the attack and its derived *Reference* class makes reference to vulnerabilities, alerts from IDSs, data about malwares, and other information from the IODEF cybersecurity extension format. The *Service* and *Application* classes describe details about attributes related to resources involved in the incident. Finally, *AdditionalData* class is included to extend the IODEF model, representing different attributes like Logs, HashFromMalware, Agent, Title, Size, IpCC, IpOrigin, TtConnections, ProxyType.

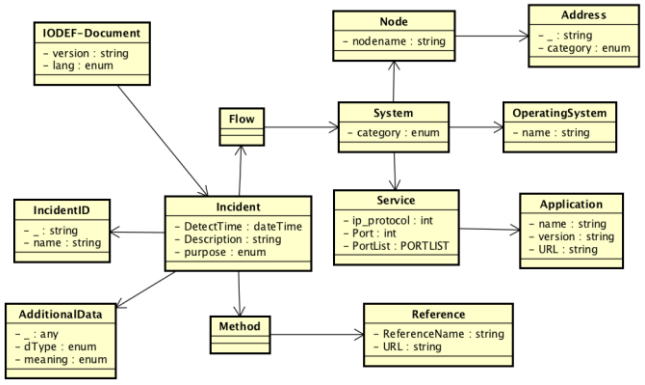

Fig. 1. The IODEF representation used by incident resolution cases.

B. Resolution of Cybersecurity Incidents

The cybersecurity incident experiences of problem-solving retrieved from the case base ought to be the most similar cases to the current problem. As implemented in the CbCSecIRS, this similarity is indicated by a numerical value between 0 and 1, where 1 is the highest similarity between two cases. The similarity computation is developed by comparing n pairs of a_i and b_i attributes represented in the case structure. Once such local similarities (similarities between attributes) are computed, a global similarity (similarities between cases) is measured. To compute this global similarity, an aggregation functions make use of weight values associated to each attribute used in the similarity computation. As described in the Equation (1), these weight value W_i represent the relative importance of the i attributes in the solution of the problem. Based on this similarity assessment, the resulting similarity computation indicates how similar the cases a and b are.

$$sim(a,b) = \sum_{i=1}^{n} W_i \times sim_i(a_i, b_i) \qquad (1)$$

To compute the distance between two cases, the Euclidean distance function is used. The solution of a cybersecurity incident problem involves the characterization of a problem situation and the consequent selection and execution of a set of actions/procedures directed to the correction (mitigation) of the problem. In this work, these actions are recorded in body of cases as simplified plan-like structures of incident treatment. Figure 2 illustrates a plan constructed by security analysts from the security division of a commercial data center to approach a Bot incident type. In practice, this plan details a cybersecurity resolution script that is followed by these analysts when they need to treat a cybersecurity incident situation.

Incident Response Plan

Incident #997164

1. Quickly disable the host access to the institution's network
2. Open a new ticket to inform the incident to the User Support Center in order to send a technician to the computer's place
3. Analyze evidence to identify the incident
4. Execute a complete scan using the antivirus software
5. If some infected file was found, follow the instructions prompted by the antivirus software
6. If the antivirus software cannot be executed, reboot the computer in "safe mode" and repeat the steps 4 and 5 of this plan and then reboot the computer in "regular mode"
7. If the antivirus software can't be executed in "regular mode" or "safe mode", use the specific tool for the infected file removal
8. If some Operation System file is infected, reinstall the Operation System
9. Ensure that the Operation System is working with the most recent updates, and the updates are set to be automatic
10. Enable the access of this host to the network of the organization
11. Ensure that the computer's firewall is installed with the most recent updates
12. Ensure that the antivirus is installed with the most recent updates
13. Recommend the user to follow orientations from the Internet Security Document available at https://cartilha.cert.br
14. Contact the Incident Security Center in order to infor them about the response given to this incident

Fig. 2. Response plan used in the treatment of a *Bot* incident.

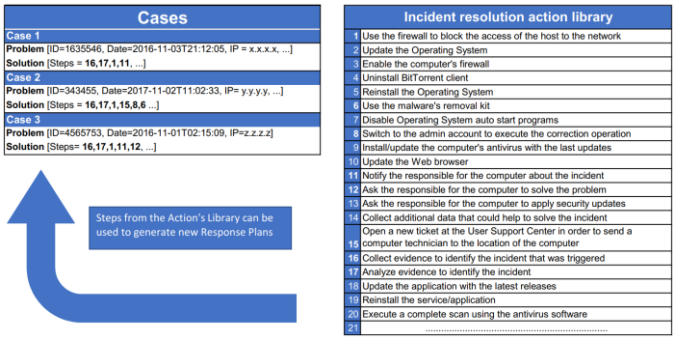

Fig. 3. Incident cases represented according to a cybersecurity incident resolution action library.

To standardize the description of cybersecurity incident resolution plans, a set of actions was represented in a library. So, resolution actions are reused from this repository in the specification of treatment plans for different kinds of cybersecurity incidents. For instance, Figure 3 presents three different cases in which their respective treatment plans were detailed according to plan step indices defined in the library (labeled according to such indices). In practice, the library reflects the steps used in the treatment of the cybersecurity incidents stored in the case base of the CbCSecIRS.

In our project, the CbCSecIRS implemented the K-Nearest Neighbours algorithm, where a (weighted) Euclidian distance function was used in the computation of case similarities, and the consequent retrieval of cases from the case base as to provide cybersecurity treatment answers to incident situations detailed as queries.

V. Experiments and Results

Experiments were developed as part of the evaluation of the CbCSecIRS approach proposed in this work. The goal was twofold: first, to assess the reuse of past experiences of cybersecurity incident problem-solving in the resolution of new problems in this cybersecurity domain and, second, to assess the accuracy of the CbCSecIRS implemented. To approach these goals, a set of 259 cybersecurity incidents used in the experiments were collected from the security division of a commercial data center.

To approach the first experimental goal, new cybersecurity incident situations were collected and used in the tests: the cybersecurity incidents number 2102389 and 2261674 (these are solved cybersecurity incident problems by different participants of the security team of the company, although they were not known during the system development). Each one of these new case problems was expressed as a query in the CbCSecIRS, allowing one to retrieve the most similar cases to them from the case base. In many senses, the aim was to examine if the cybersecurity resolution procedures recorded in the retrieved cases could be reused on the treatment of the current problem. In doing so, the retrieved cases for each executed query were presented to a security expert from the commercial data center organization. Whenever possible, this expert offered positive feedback when the resolution plan retrieved could be properly reused on the treatment of the current problem situation. An example of such research in action case study is presented in Figure 4.

Incident			
ID 2102389			
Incident		Solution steps	
DetecTime	2017-05-09 14:44:30	1	16
IP	21	2	17
Logs	...	3	28
Description	...	4	29
Type	Bot	5	8
RefID	-	6	6
Category	Source	7	9
Port	34934	8	2
Hostname	26	9	3
IpCC	65	10	31
TtConnections	-	11	38
Protocol	http	12	23
MalwareName	Downadup	13	-
RefURL	36	14	-

Recommended solution #1			
ID 1483711			
Incident		Solution steps	
DetecTime	2016-09-01 12:21:31	1	16
IP	23	2	17
Logs	...	3	28
Description	...	4	29
Type	Bot	5	8
RefID	-	6	6
Category	Source	7	9
Port	34590	8	2
Hostname	28	9	3
IpCC	65	10	31
TtConnections	-	11	38
Protocol	http	12	23
MalwareName	Downadup	13	-
RefURL	36	14	-

Recommended solution #2			
ID 1510754			
Incident		Solution steps	
DetecTime	13-09-16 13:20:00	1	16
IP	23	2	17
Logs	...	3	28
Description	...	4	29
Type	Bot	5	8
RefID	-	6	6
Category	Source	7	9
Port	42247	8	2
Hostname	28	9	3
IpCC	65	10	31
TtConnections	-	11	38
Protocol	http	12	23
MalwareName	Downadup	13	-
RefURL	36	14	-

Fig. 4. Incidents number 2102389, 1483711 and 1510754.

In Figure 4, the 2102389 incident was used as a query in the CbCSecIRS, allowing one to retrieve the 1483711 and 1510754 incidents from the case base. All these incidents were characterized as Bot types. These retrieved cases have treatment plans that were considered similar to the plan recorded in the query case. So, the CbCSecIRS was successful on the resolution of this 2102389 test case, showing that the proposed technique was able to maintain the cybersecurity incident resolution knowledge to this kind of problem.

Another example is presented in Figure 5. To the 2261674 incident used as query, the 1022675 and 1620589 cases were retrieved from the CbCSecIRS case base. Both retrieved cases were of the *Copyright* type detailing the illegal sharing of movies in the *BitTorrent* platform. In relation to the treatment plan represented in the retrieved cases, only the 1022675 case contained a highly similar treatment plan in relation to the plan recorded in the query case. Although a solution to the 2261674 query situation could be obtained with the reuse of the plan recorded in the most similar case retrieved, the 1620589 case recorded a new kind of treatment in relation to the other cases considered. Figure 5 presents these cybersecurity incident treatment plans side-by-side, allowing one to observe that the 2261674 incident contained more detailed resolution steps than the more general resolution ones represented in 1620589 case. It means that the retrieved solution could not be fully reused in the solution of the test case situation. That was because it was necessary to develop more particular resolution actions in the treatment of the current problem situation. All in all, as part of traditional knowledge acquisition and representation tasks, improvements in the ways cybersecurity resolution procedures are represented in cases still have to be applied in the CbCSecIRS proposal.

IdIncident	2261674	1620589
Step 1	Collect evidences to identify the incident that was triggered	Collect evidences to identify the incident that was triggered
Step 2	Analyze evidences to identify the incident	Analyze evidences to identify the incident
Step 3	Use the firewall to block the access of the host to the network	Use the firewall to block the access of the host to the network
Step 4	Open a new ticket at the User Support Center in order to send a computer technician to the location of the computer	Notify the administrator of host/network/server/webpage about received incident
Step 5	Uninstall the client of BitTorrent protocol	Request the administrator of the host/network/server/webpage to block the communication with the network
Step 6	Guide the user to follow the published security tips	After incident treatment, request access
Step 7	After incident treatment, request access	After incident resolution, unblock the access in the firewall
Step 8	After incident resolution, unblock the access in the firewall	Notify the CAIS about incident resolution
Step 9	Notify the CAIS about incident resolution	

Fig. 5. Incident resolution plans for cases 2261674 and 1620589.

In addition to such research in action case study experiments, tests aiming to evaluate the CbCSecIRS accuracy were developed as part of the second experimental goal. To do so, the cases in the case base were randomly divided in p partitions of equal size, where $p = 10$. Then, a *K-Fold Cross Validation* technique was used in the evaluation of the system accuracy. In different test runs, for instance, the cases belonging to one of these partitions were used as query cases, while the remaining cases were maintained in the case base so that they could be retrieved as solutions for such a query. In case the retrieved cases and the query cases contained similar cybersecurity incident resolution plans, the answer generated by the system was considered correct. Otherwise, the system offered an incorrect answer to the current problem situation. In

a first run, tests were developed using a similarity function in which a weight = 1.0 was attached to all case attributes being used in the similarity computations, indicating that such attributes have the same importance in such computations. In a second run, the weight values for such case attributes were adjusted according to the opinion of a cybersecurity domain expert from the commercial data center organization.

Table II shows the accuracy results obtained when the K-Fold Cross Validation technique was executed. Although considering different similarity thresholds in the retrieval algorithm used by the CbCSecIRS (95% and 60% minimal similarities), these accuracy results were positive (i.e. as good as to accuracy results presented by other works in this application domain [11-14]) when adjusted weight values were used and when all weight values were equal to 1.0 in the similarity function used by this system.

TABLE II. THE ACCURACY OF THE CBCSECIRS

	1-NN	2-NN	3-NN	4-NN	5-NN
Similarity threshold = 60%, weights w = 1	87.50	84.38	88.89	83.33	80.00
Similarity threshold = 95%, weights values determined by a domain expert	93.33	90.00	95.24	91.67	90.00

VI. CONCLUDING REMARKS

Organizations spent a lot of time and money on the treatment of cybersecurity incidents due to the fact that it is still challenging to maintain their concrete experiences of cybersecurity problem-solving. To approach this problem, this work describes the knowledge acquisition and representation activities that cybersecurity system developers can explore when building CbCSecIRS. In doing so, the cybersecurity incident case model used by these CbCSecIRSs is based on attributes detailed in the IODEF standard. Instead of acting as an isolate cybersecurity solution, the overall idea of following the IODEF standard is to permit to integrate the reasoning capabilities from both intrusion detection systems and cybersecurity incident resolution systems.

As discussed in this work, the CbCSecIRS offers the capability of retrieving cybersecurity incident data and incident resolution procedures represented in cases. Such cybersecurity knowledge is are organized and specified explicitly in the case structure, allowing to be reused by security analysts in different cybersecurity problems. In particular, cybersecurity knowledge regarding incident resolution actions now recorded in cases amount to a concrete explanation about how to better approach those kinds of problems. This explanation capability is crucial when cybersecurity emergency circumstances occur (i.e. after an attack happened, even in face of protection barriers). That is because security analysts are required to promptly and effectively explain their actions in such crisis situations as to mitigate the damage that a cybersecurity event very often causes in the computer infrastructure of an organization.

The CbCSecIRS proposal detailed in this work can have a dual application since it can be explored in both the cybersecurity incident detection and the cybersecurity incident resolution. In practice, cybersecurity incident cases do express incident resolution knowledge which can complement the functionalities required to automatically detect and prevent those incidents as explored by other AI techniques in the cybersecurity domain. Although the experiments presented here can be expanded in different ways, the results show a positive scenario in which our CbCSecIRS proposal is relevant for cybersecurity analysts because it accurately relies on similarity-based computations to connect incident detection data with incident resolution procedures which can now be maintained in the structure of reusable cases.

REFERENCES

[1] ISO/IEC, "ISO/IEC 27035:2016, Information technology - security techniques - information security incident management," Int. Organization for Standardization, 2016.

[2] R. Danyliw, "RFC 7970: The Incident Object Description Exchange Format Version 2," Internet Engineering Task Force (IETF), 2016.

[3] T. Takahashi, K. Landfield, and Y. Kadobayashi, "RFC 7203: An Incident Object Description Exchange Format (IODEF) Extension for Structured Cybersecurity Information," Internet Engineering Task Force (IETF), 2014.

[4] T. Takahashi, and D. Miyamoto, "Structured cybersecurity information exchange for streamlining incident response operations," in NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, Istanbul, Turkey, 2016, pp. 949-954.

[5] H. Gascon, B. Grobauer, T. Schreck, L. Rist, D. Arp, and K. Rieck, "Mining attributed graphs for threat intelligence," in Seventh ACM on Conf. on Data and Application Security and Privacy (CODASPY '17), Scottsdale, Arizona, 2017, pp. 15-22.

[6] M. B. Line, I. A. Tøndel, and M. G. Jaatun, "Current practices and challenges in industrial control organizations regarding information security incident management – Does size matter? Information security incident management in large and small industrial control organizations," *Int. Journal of Critical Infrastructure Protection,* vol. 12, pp. 12-26, 2016.

[7] R. L. d. Mantaras, D. McSherry, D. Bridge, D. Leake, B. Smyth, S. Craw, B. Faltings, M. L. Maher, M. T. Cox, K. Forbus, M. Keane, A. Aamodt, and I. Watson, "Retrieval, reuse, revision and retention in case-based reasoning," *The Knowledge Engineering Review,* vol. 20, no. 3, pp. 215-240, 2005.

[8] K. D. Althoff, and R. O. Weber, "Knowledge management in case-based reasoning," *The Knowledge Engineering Review,* vol. 20, no. 3, pp. 305–310, 2005.

[9] K. Dalkir, and J. Liebowitz, *Knowledge Management in Theory and Practice*: The MIT Press, 2011.

[10] S. Metzger, W. Hommel, and H. Reiser, "Integrated security incident management – concepts and real-world experiences," in Sixth Int. Conf. on IT Security Incident Management and IT Forensics, Stuttgart, Germany, 2011, pp. 107-121.

[11] F. Jiang, T. Gu, L. Chang, and Z. Xu, "Case Retrieval for Network Security Emergency Response Based on Description Logic," in 8th Int. Conf. on Intelligent Information Processing (IIP), Hangzhou, China, 2014, pp. 284-293.

[12] H. K. Kim, K. H. Im, and S. C. Park, "DSS for computer security incident response applying CBR and collaborative response," *Expert Systems with Applications,* vol. 37, no. 1, pp. 852-870, 2010.

[13] L. Ping, Y. Haifeng, and M. Guoqing, "An incident response decision support system based on CBR and ontology," in Int. Conf. on Computer Application and System Modeling (ICCASM 2010), Shanxi, Taiyuan, 2010, pp. 337-340.

[14] G. Capuzzi, L. Spalazzi, and F. Pagliarecci, "IRSS: Incident Response Support System," in Int. Symposium on Collaborative Technologies and Systems (CTS 2006), Las Vegas, NV, USA, 2006, pp. 81-88.

[15] A. Phillips, and M. Davis, "RFC 4646: Tags for Identifying Languages," Network Working Group, 2006.