An Intelligent Forensic Framework towards Cloud: Its Ontological Aspects

Suchana Datta
Department of Computer Science & Engineering
MAKAUT, WB (formerly known as WBUT)
Salt Lake City, Kolkata, India

Chandan Pan
Big Data Analyst
Ex-Student (Praxis Business School)
Kolkata, India

ABSTRACT

Cloud Computing, a relatively new concept and all its associated methodologies offer uncountable advantages now-a-days. These advantages range from integrating different systems, offering guarantee over searching mean distribution and to software tools integration, used by various cloud service providers and consumers. So all these provisions are not only making our lives easier but attract lots of intruders and malicious actors to perform various cloud crimes. This paper aims to contribute towards the design of an ontology based cloud forensic framework with a view to identify the malicious actors. The proposed framework consists of mainly two components - Ontology-Enabled Forensic Blackboard (OFB) and Ontology-Enabled Forensic Controller and Processor (OFCP). The main function of OFB is to communicate with the investigators after receiving the classified crime incident scene collected from VM snapshots where ontology base is used spontaneously to distribute the investigators' request for proper information relevant to the investigation. Whereas, the function of the OFCP is to interact with different Cloud Malicious Actor Identifier (CMAI) so that accurate information can be gathered based on the distributed request with the help of a meta-ontology framework that acquire and restructure data using different AI reasoning tools and finally the mapping with its corresponding requests is done.

General Terms

Cloud Forensics, Ontology, OWL, Knowledge Base

Keywords

Digital forensics; cloud computing; cloud forensics; SaaS; PaaS; IaaS; virtualization; Ontology; OWL; Protégé;

1. INTRODUCTION

Researchers formulated Cloud Computing as a general term through which a class of on-demand complex computing services has been illustrated. Various commercial vendors, like- Google, Amazon and Microsoft introduced this new challenging technology. A model is denoted by it in such a way that the infrastructure of computing acts as a "Cloud", beyond users physical reach. Thus companies as well as individuals are able to access all applications based on demands throughout the whole world. Providing storage, computing and software "as a service" is the main motto behind this extensively scalable computing platform [1].

The dynamic and on-demand provisions of cloud computing is making attackers more and more prone to this new technology. And consequently, Cloud Forensics has been introduced. Cloud Forensics [2] is a relatively new discipline that investigates cloud crimes by collecting and analyzing malicious activity details. Extensive distributive nature of it and coupling with the massive amount of data is making the automation of cloud forensics gradually a necessary one. Automating the cloud forensics process demands a set of systematic analysis techniques and existing expert knowledge to be converted into an intelligent decision-making and analysis system [3].

In this respect, Ontologies play a great role especially in intelligent knowledge representation. Whenever any cloud crime occurs, investigators start investigating with the help of Cloud Service Providers of different cloud systems. Classifying the incident scene from the collected VM snapshots, searching for relevant information based on that classification and finally collecting proper evidences against the malicious actors (CSPs or cloud consumers or intruders), all demand proper knowledge and accuracy so that the investigation can be carried out reliably and with intelligence.

Keeping this in mind, in this paper an ontology based cloud forensic framework has been introduced that discovers the investigators' interoperability with Cloud Malicious Actor Identifiers. The proposed model mainly consists of two components. Ontology has been used by both the components in discovering the best response both from the investigator and Malicious Actor Identifiers' end. As the paper moves forward, the background of this proposed framework has been enlighten. Then the proposed ontological cloud forensic model has been illustrated with proper system architecture, explanation and the algorithm and finally the paper ends up with future research direction for the proposed forensics system.

2. BACKGROUND & PRELIMINARIES

2.1 Cloud Forensics

Cloud Computing environment offers users three most common service models, IaaS (the equipment for supporting operations are outsourced by the service providers), SaaS (customers are entertained the software provisions only), PaaS (the customers are allowed to use not only operating systems but virtualized servers too provided by the service providers which helps to run applications over the internet with all associated services). Cloud Forensic can be defined as the cross discipline of Digital Forensics and Cloud Computing [4]. Digital Forensic is the branch of science where various

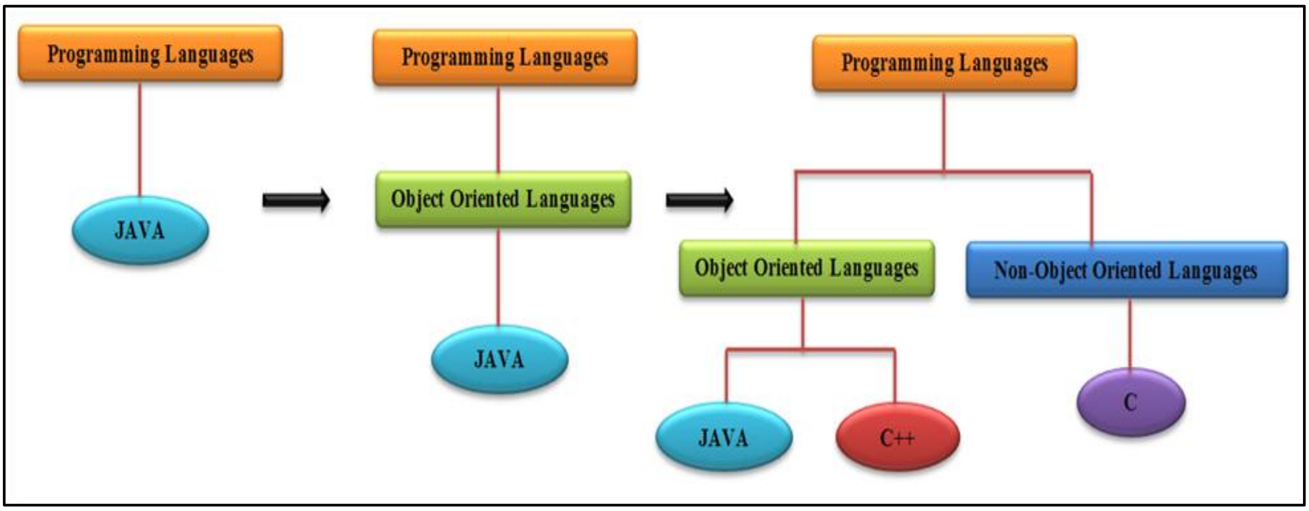

Programming Languages
JAVA
Programming Languages
Object Oriented Languages
JAVA
Programming Languages
Object Oriented Languages
Non-Object Oriented Languages
JAVA
C++
C

Fig 1: Ontology of programming languages

computer science principles are applied for the purpose of recovering electronic evidences and presenting them in the court of law. Cloud Forensic can be expressed as the subset of Network Forensics that mainly concerned with the forensic investigation of networks [5-6]. Broad network access is the base of cloud forensic. Therefore all the basic phases relevant to the network forensics are maintained by cloud forensics but with the tailor with cloud computing environment. There are lots of complex aspects associated with cloud computing despite ensuring service availability and cost-effectiveness. Segregation of duties are provisioned amongst CSPs and customers, at the same time interaction among multiple tenants is also mandatory those are sharing same cloud. As the paper moves forward, the background of this proposed framework has been enlighten. Then the proposed ontological cloud forensic model has been illustrated with proper system architecture, explanation and the algorithm and finally the paper ends up with future research direction for the proposed forensics system.

2.2 Ontology

Despite being a concept of philosophy [7] initially, researchers adopted ontological usage within the domain of information system. Artificial Intelligence and Knowledge Management welcomed ontology in a broader aspect because it has the ability to filter massive amount of data. Gruber [8] defined ontology as “an explicit specification of a conceptualization”. Ding and Foo quoted this definition and declared ontology as the term that refers to understanding and sharing some domain of interest [9]. According to them, ontology is something that “often conceived as a set of classes, concepts, relations, functions, axioms and instances” [10]. Gruber raised this concept in his work [11] where it has been declared that various object definitions illustrate ontology. Only those objects are to be considered those exist with the domain of programs. Names of various objects and texts are associated with such definitions those elaborate the meaning of the name in an understandable manner to humans. Fig-1 illustrates the ontology of programming languages.

The basic relationship amongst these objects has been shown as basic hierarchy. A very small part of programming language has been formed in the form of a hyper tree. Souza et al. presented this as a tool for ontology visualization [12]. As ontology represents something from the real world accurately, consequently it is expected that hierarchy of ontology owes accuracy [13]. First, domain of ontology is considered. The first question that is asked is what the domain is and what its scope is. As in the example described in the Fig–1, the scope of the programming languages might demand to be narrowed with respect to the domain of all object oriented programming languages. In this example, it should be stated by ontology of programming language that JAVA is a programming language where it has been represented the JAVA object as the subclass of programming language. A further refinement can be added stating that JAVA is an object oriented programming language. Here in this ontology, as the knowledge base goes on enriching, it can be inferred that C++ is also an object oriented programming language but not C. The domain of all ontology must be represented through a hierarchical tree structure where a set of competency questions are asked to conceptualize and formalize the ontology as it has been shown in the fig-1.

2.3 Ontology Development Life Cycle

Prior to the designing of any cloud forensics ontology, it is recommended to use a hyper approach where different key features of different approaches for ontology development are combined together so that the knowledge bases become stronger. Few very familiar approaches, like-METHONTOLOGY [14], method proposed by Uschold and King [15] and the approach of Gruninger and Fox [16] can be incorporated into this hyper model of ontology building which gives better performance in creating an established knowledge base. Three main interdependent phases are to be maintained throughout the life cycle of any ontology- Specification, Conceptualization and finally Formalization-Implementation.

2.3.1 Specification

Before designing any ontology, it must be formalized and granulized mentioning its domain of interest. Description logic plays a great role in ontology knowledge representation paradigm where formalizing the knowledge is allowed. In this paper, the ontology of cloud forensics has been designed that represents the knowledge about cloud forensics domain. Concepts, their relationship, attributes and facts about these concepts are incorporated while building up the concerned ontology. Additionally, like method ontology, the usage of domain knowledge in case of solving problems or complex reasoning. The proposed ontology includes knowledge that represents the cloud forensics domain as well as the investigation process of cloud crime.

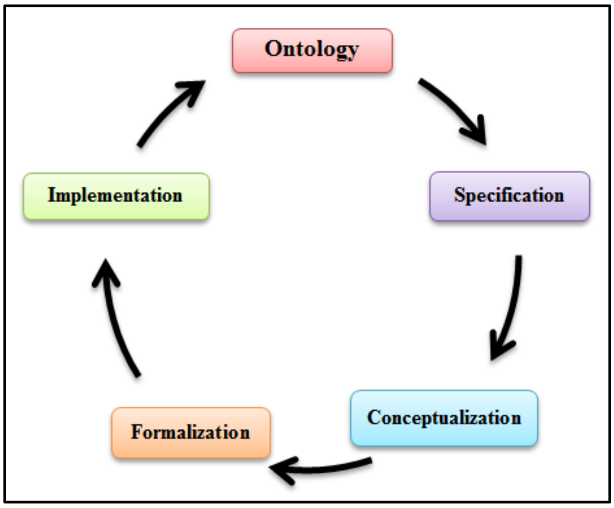


Fig 2: Ontology development life cycle

2.3.2 Conceptualization

After specifying the domain of interest regarding building up the ontology, the next step is to conceptualize it. In this stage, the basic concepts of classes are identified and must be relevant to the concerned domain. In most of the ontology conceptualization, three basic types of knowledge can be represented, like- goal of the problem solving, its knowledge while solving the problem and factual knowledge about the particular domain. Usually by a hierarchical tree or taxonomic structure, a group of competency questions are structured. The main feature of this kind of structure is any answer to any parent competency question demands proper answer from all its child competency questions available. These set of competency questions are the key to identification of any goal towards solving any defined problem. Competency questions and their corresponding answers play major role in case of acquiring knowledge regarding the problem scope submitted. All the necessary constraints input and output data are also assumed.

2.3.3 Formalization & Implementation

The last stage of the life cycle of any hyper ontology model is to formalize the concept and its suitable implementation. With the help of appropriate ontology language, the concept is formalized. Then the final stage comes which is the actual implementation of the formalized knowledge for the respective ontology and feeding it with individual class instances. Lots of description logic languages can be used for the purpose of implementation. Web Ontology Language (OWL) can be used as the encoding language. OWL is nothing but a description logic based language that is used not only for developing ontologies but representing knowledge through semantic web also.

3. RELATED WORKS

The use of ontology over cloud architecture has been proposed by Lamia Youseff et al. [17] where a very good effort to establish a knowledge base is can be observed in the domain of cloud computing. Cloud ontology has been portrayed there as a stack of layers. They discussed the strong points of each layer and issues related to the integration and communication amongst layers has been neglected.

The use of ontology was proposed for the first time by Raskin and Nirenburg in 2001 [18] for computers and information security. They primarily focused upon highlighting mainly what are the advantages can be entertained by the domain of information security using ontology. Currently, security tools, like- Intrusion Detection System, antivirus, Malware Detection Systems are also being developed with the help of ontology.

An ontological framework has been introduced by N. M. Karia et al [19], providing different cloud environment structures and description of their components. But the work fails to address the heterogeneity (vendors and number of standards).

Achieving ontology concept in several services of cloud has been attempted by Teodor-Forin Forties et al. [20]. mOSAIC is introduced by the author as a middleware, so that the communication can be facilitated by this amongst various cloud providers. The use of ontology has been suggested not only in selection and execution but in discovery of services and resources too. A comprehensive model based on ontology has been introduced only for public clouds.

A Cloud Service Discovery System (CSDS) has been presented by Tackgyeang Han and KwangMangSim on ontology [21]. Ontology has been used for enhancing the performance of the system mainly. The concept of software agent is used which consults ontology while information retrieval from public cloud.

Miranda Zang et al. [22] described the use of ontology on Cloud Based IaaS Services. Focusing on IaaS services, ontology has been implemented in the Cloud Recommended System by the author. PaaS and SaaS configurations were neglected in a public cloud. In order to enhance the performance of cloud services, an agent-based support system framework has been proposed by Tashihiro Uchibayashi et al. [23] so that public cloud services can be discovered. Author used JADE based agents in order to measure the usage of network based on the information of user requirements.

4. PROPOSED FRAMEWORK

4.1 System Design

In this paper, the structure and work flow of an ontology-based cloud forensic framework has been introduced. This framework aims to discover the cloud malicious actors' interoperability in the cloud system. These actors can be any cloud consumer or service provider or any external third party malicious user. In digital forensics, whenever any malicious activity is reported, investigators investigate the entire log records of the victim system and thus ultimately find out the reason behind the problem and the investigation report along with the evidence proof is presented in the court of law in a proper and authenticated way. But with the advent of cloud computing, the scenario of traditional digital forensics demands to be changed so that the digital forensic science can cope up with the cloud distributed architecture and also can come up with proper and authenticated forensic investigation report as it was used to be done in traditional digital forensic system. Though the distributed nature of cloud provides numerous advantages to consumers, but when a malicious

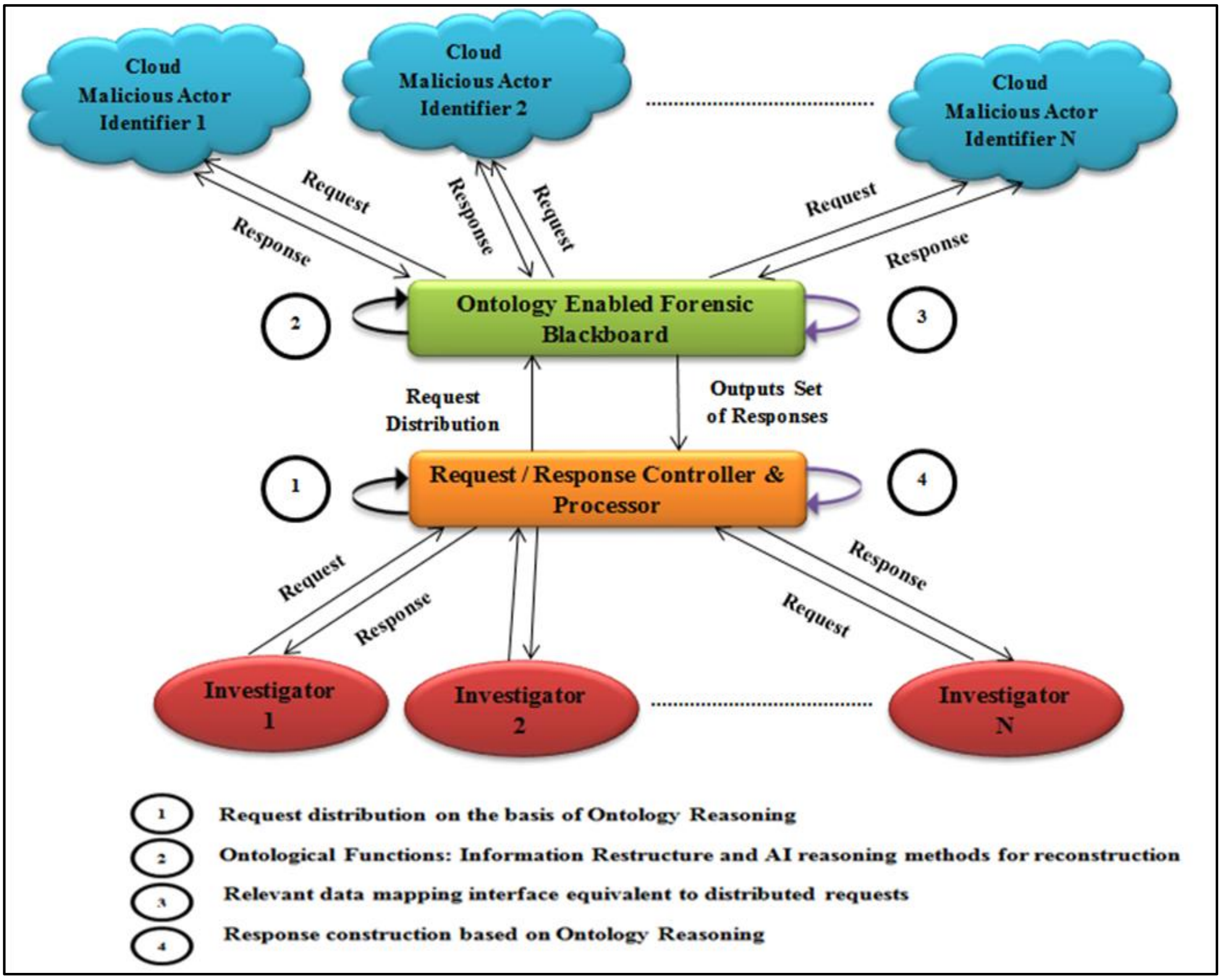
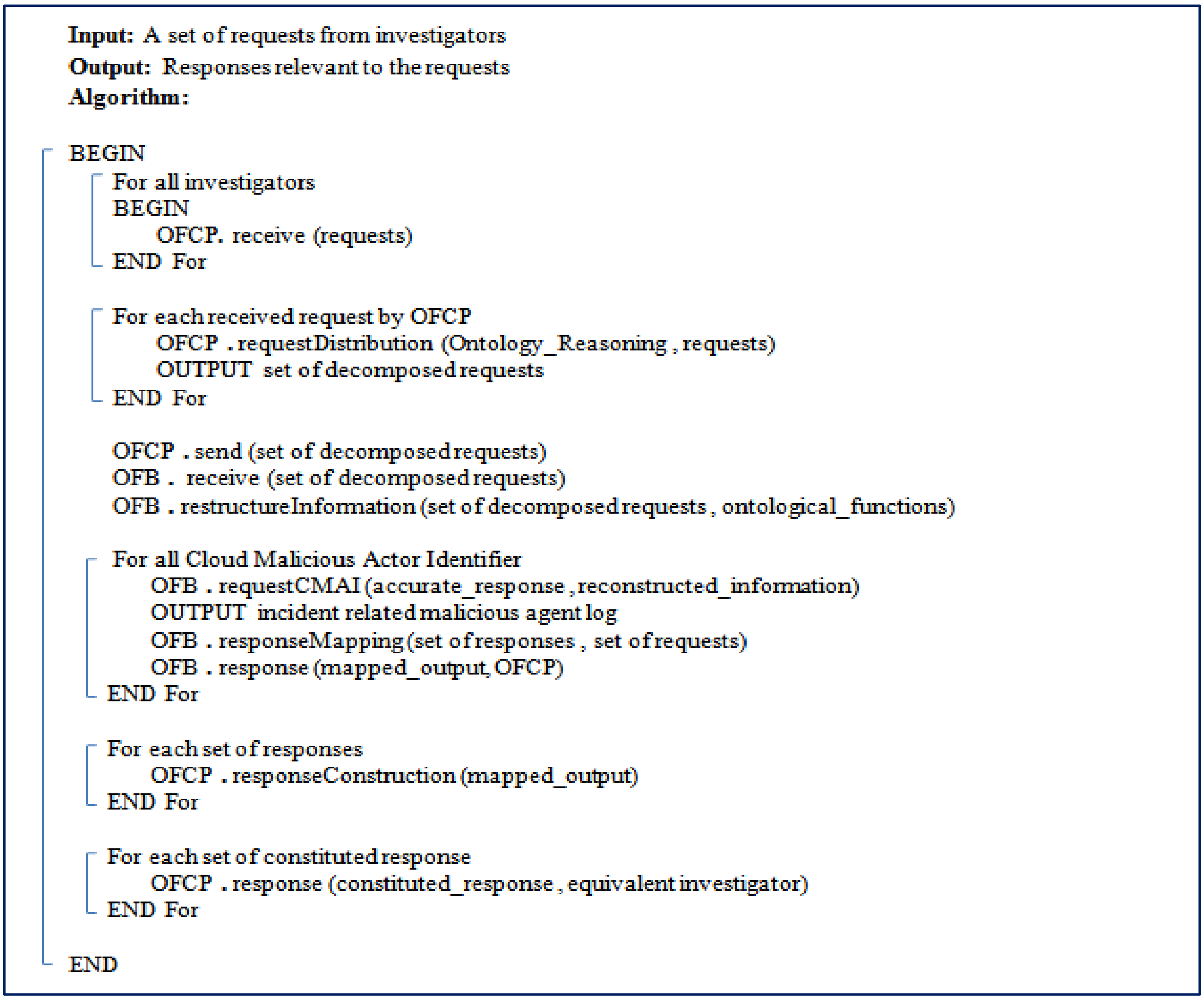

Cloud Malicious Actor Identifier 1
Cloud Malicious Actor Identifier 2
Cloud Malicious Actor Identifier N
Request
Response
Response
Request
Request
Response
2
Ontology Enabled Forensic Blackboard
3
Request Distribution
Outputs Set of Responses
1
Request / Response Controller & Processor
4
Request
Response
Response
Request
Investigator 1
Investigator 2
Investigator N
1 Request distribution on the basis of Ontology Reasoning
2 Ontological Functions: Information Restructure and AI reasoning methods for reconstruction
3 Relevant data mapping interface equivalent to distributed requests
4 Response construction based on Ontology Reasoning

Fig 3: Architecture of the proposed model

activity is reported to any cloud service provider, the situation becomes a menace. It becomes very challenging to trace the malicious actors and identify them with accuracy and authenticity from numerous systems and their logs collected by investigators. Moreover after classifying the crime scene from the collected VM (Virtual Machine) snapshots provided by VMM (Virtual Machine Manager), it is another challenge to propagate the request of appropriate investigator for relevant log data. And if this investigation is thought to be done manually, it is something next to impossible.

So there is a need of any automated system which will propagate the investigators' request to proper Malicious Actor Identifier in one hand, on the other hand Malicious Actor Identifiers are also supposed to collect the relevant data through an automated system. In this proposed model, this automation is represented in an ontological aspect where after each and every complete investigation process, the knowledge base of the proposed framework will be enriched by the ontological analysis and reasoning that an investigator would have gone through during his / her investigation process. The knowledge bases of Malicious Actor Identifier system will also be populated simultaneously so that in future, whenever same kind of crime incident will get reported by any user, the investigator can get full support of experience of the proposed forensic system consulting its knowledge base as and when required. Both the knowledge bases will again be populated if any investigator came across any new situation, new crime scene, new attack or new evidence and proof of concept. A blackboard design style has been adopted in designing the proposed forensic model. Two main components have been taken up for building up this framework- Ontology-Enabled Forensic Blackboard (OFB) and Ontology-Enabled Forensic Controller and Processor (OFCP) as shown in Fig–2. OFCP can directly communicate with cloud forensic investigators. As soon as it receives the investigators request for acquiring proper log record to identify the malicious actors, the functionality of OFCP starts. OFCP-ontology which is in built decomposes the investigators' request into a set of mini-requests (distribution of primary basic components). Hardly had it receives any response from the OFB component when the final process by OFCP is started by constructing the set of mini responses into a single consistent response and the response is sent back to the accurate communicated investigator.

On the other hand, OFB component interacts with not only all possible Cloud Malicious Actor Identifier but various agent systems and OFCP controller too. With the help of meta-ontology it reconstructs the distributed request data. It maps



```
Input: A set of requests from investigators
Output: Responses relevant to the requests
Algorithm:

BEGIN
    For all investigators
    BEGIN
        OFCP. receive (requests)
    END For

    For each received request by OFCP
        OFCP . requestDistribution (Ontology_Reasoning , requests)
        OUTPUT set of decomposed requests
    END For

    OFCP . send (set of decomposed requests)
    OFB . receive (set of decomposed requests)
    OFB . restructureInformation (set of decomposed requests , ontological_functions)

    For all Cloud Malicious Actor Identifier
        OFB . requestCMAI (accurate_response , reconstructed_information)
        OUTPUT incident related malicious agent log
        OFB . responseMapping (set of responses , set of requests)
        OFB . response (mapped_output, OFCP)
    END For

    For each set of responses
        OFCP . responseConstruction (mapped_output)
    END For

    For each set of constituted response
        OFCP . response (constituted_response , equivalent investigator)
    END For

END
```

Fig 4: Algorithm of the proposed framework

data to its equivalent redistributed requests using plenty of AI reasoning methods. Initially distributed set of requests are received from the OFCP by OFB system and then the reconstruction process begins by means of numerous AI reasoning methods consulting the meta-ontology. Changes are updated immediately during runtime by populating the corresponding knowledge base. Forward or backward reasoning is used to justify the acceptable responses that are got from the streaming flow of data between OFB and external agents. At this stage the OFB processes get started the mapping of requests and their equivalent responses. At the end, the response set and the corresponding mapping parameters are sent to the OFCP.

5. ONTOLOGY-ENABLED MODEL DESCRIPTION

The information that is under consideration of analysis varies usually from case to case in case of digital forensics. So there must be some automated process which is able to populate the ontology dynamically. In the proposed model, Information Retrieval Unit retrieves the relevant information to the queries of Ontology-Enabled Forensic Blackboard (OFB) as well as the Ontology-Enabled Forensic Controller and Processor (OFCP) from the large data storage with the help of Malicious Actor Identifiers. All the results are presented to the proper investigators besides populating the ontology for future consultation. This extraction is done by the Ontology Populate Unit from the Information retrieval Unit. And finally the unit of knowledge base infers new knowledge. This knowledge inference is done mainly by an inference engine, firing different inference rules. Domain specific queries of both OFB and OFCP are met up by the Ontology Query and Response Unit.

5.1 Ontology Design

Ontology is nothing but the 'specification of conceptualization'. Some of the concepts of cloud forensic rather Malicious Actor Identifiers and the relevant malicious data, characteristics of those concepts and relationships amongst them are to be conceptualized initially. All these details can be presented in an ontological form formally. OWL (Web Ontology Language) [24-25] is preferred here as the ontology representation. This ontology is supposed to be developed using a tool named as Protégé [26]. Once the ontology is designed, incident relevant information is extracted in order to populate the ontology.

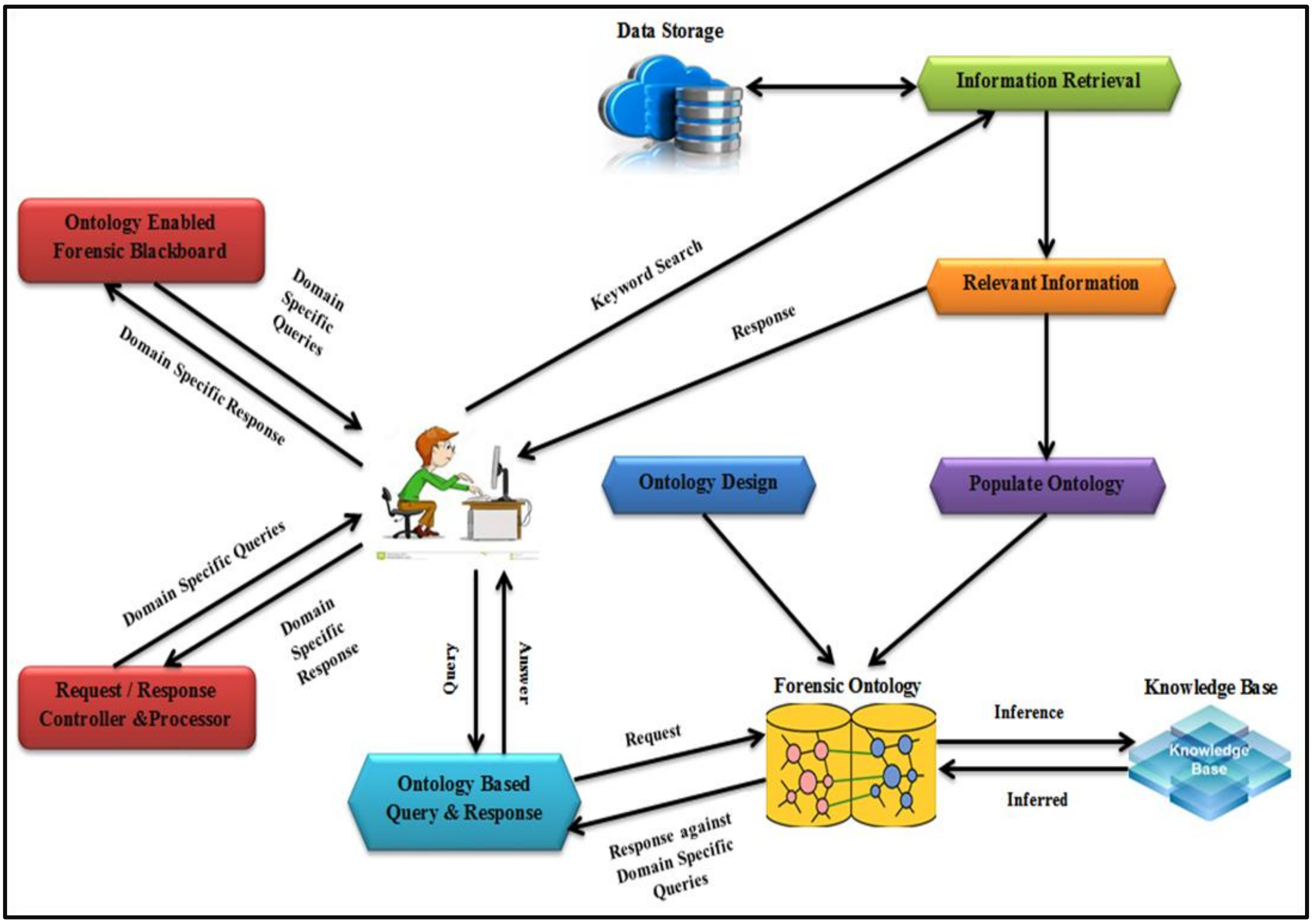


Fig 5: Ontology-enabled proposed system description

5.2 Retrieve Information & Populate Ontology

Using several information retrieval techniques [27], relevant malicious data and the corresponding details of the actors are retrieved from the large data storage. All the related information is to be parsed, analyzed, tokenized and indexed so that a highly efficient cross-reference lookup can be facilitated for a rapid search. All the investigators' query keywords are first analyzed, then refined and mapped against those indexes so that the classified incident related information can be retrieved very easily and efficiently. Retrieved information will then be ranked according to the relevance of the query given by the investigators and thus they will get proper information based on their requirement for the investigation. Ontology will also be populated besides meeting up with the investigators' queries. For the purpose of ontology population, the "Data-Master" plug-in provided by Protégé can be used [28].

5.3 Ontology Query & Response

SPARQL query language is capable of developing ontology queries. In the proposed framework, the domain specific queries those can be structured for the ontology, like:

a) Is the retrieved information data set is relevant to the classified incident scene collected by the VMM?

b) Are all the attributes of the objects of the data set are relevant to the crime scene?

These questions can be answered by the ontology. Suppose, if the incident that is to be investigated is network related, then the following queries can enrich the knowledge base of the corresponding ontology, like-

c) What is the normal proportion of the rate of transfer of data using a particular protocol to any particular consumer?

d) Are there any deviations / discrepancies of this proportion during the period of crime incident?

5.4 Knowledge Base

SWRL (Semantic Web Rule Language), a widely used rules representing language [28] develops the inference rule relevant to the cloud forensics. New knowledge can be inferred with the help of JESS inference engine. The rules are mapped against the ontology and are fired accordingly so that new knowledge can be inferred [29]. With the help of this inferred knowledge, ontology is updated. Other rules may also be fired by this inferred knowledge. Thus an iterative fashion is incorporated in case of updating the existing ontology and as a result, some ontology becomes inconsistent. In order to get rid of this problem, consistency is checked after each and every update through Pallet Reasoner [30].

6. FUTURE RESEARCH DIRECTIONS

The proposed framework is an approach towards mitigating the problem of investigating cloud crime incident manually rather giving it an ability of automation. The use of the concept of ontology makes this automation faster and accurate. The existence of both the knowledge bases owing by both the model components make the investigation an easier one. The knowledge base of OFCP helps the investigator to propagate correct request for evidences to the correct Malicious Actor Identifier who ultimately collects the relevant evidences and information to the reported crime scene by the investigator. Populating this knowledge base each and every time makes these knowledge bases more and more enriched. This is true from the Malicious Actor Identifiers' end too. From the plenty of users, service providers and malicious actors, this automated system identifies the relevant responsible virtual machines and collect data accordingly as per the demand of investigators and populate the ontology for each and every cases. As a future research it is desirable to implement the proposed framework using SPARQL and OWL so that the flavor of ontology can be achieved. Besides doing this, Quality of Service (QoS) criteria are also taken into consideration as the future research in case of measuring the response time, throughput and availability both from the investigator and Malicious Actor Identifiers' end.

7. CONCLUSION

Cloud forensics is a branch of science where the traditional digital forensics science and the challenging and mysterious cloud computing's black box architecture has been blended in such a way that despite having extensive distributed architecture, each and every malicious actor is punished in a proper and justifiable way. In this paper, besides exploring the idea of ontology and cloud forensics, an automated ontology based forensic model has been proposed with proper system design and algorithm of the system. The entity relations involved into the model has been mentioned well so that the desired knowledge base and the domain ontologies can be developed in a suitable manner. In future the proposed system has been planned to implement with proper QoS and availability measurement factor so that the investigation process can be carried out smoothly and accurately in the cloud environment with less time and effort.

8. REFERENCES

[1] Amr Tolba, and Ahmed Ghoneim "IABCF Smarter: An Intelligent Agent IJCST Vol. 3, Issue 4, Oct - Dec 2012.

[2] Accorsi, Rafael, and Keyun Ruan. "Challenges of cloud forensics: A survey of the missing capabilities." ERCIM News 2012, no. 90 (2012).

[3] Simou, Stavros, Christos Kalloniatis, Evangelia Kavakli, and Stefanos Gritzalis. "Cloud forensics: identifying the major issues and challenges." In Advanced Information Systems Engineering, pp. 271-284. Springer International Publishing, 2014.

[4] Guo, Hong, Bo Jin, and Ting Shang. "Forensic investigations in cloud environments." In Computer Science and Information Processing (CSIP), 2012 International Conference on, pp. 248-251. IEEE, 2012.

[5] K. Kent, S. Chevalier, T. Grance and H. Dang, Guide to Integrating Forensic Techniques into Incident Response, Special Publication 800-86, National Institute of Standards and Technology, Gaithersburg, Maryland, 2006.

[6] EurActiv, Cloud computing: A legal maze for Europe, Brussels,Belgium(www.euractiv.com/en/innovation/cloud-computinglegal-maze-europe-linksdossier-502073), 2011.

[7] Noy, N. F. & McGuinness, D. L. (2001), "Ontology Development 101: A Guide to Creating Your First Ontology", Stanford Knowledge Systems Laboratory, 14th May 2005.

[8] Gruber, T. R. (1993), "A Transition Approach to Portable Ontology Specifications", Knowledge Acquisition, 5(2), 199-220.

[9] Ding, Y. & Foo, S. (2002), "Ontology Research and Development. Part I – A Review of Ontology Generation", Journal of Information Science, 28(2), 123-136.

[10] Holsapple, C. & Joshi, K. D. (2002), "A Collaborative Approach to Ontology Design", Communications of the ACM, 45(2), 42-47.

[11] Mahalingam, K. & Huhns, M. N. (1997), "A Tool for Organising Web Information", Computer, 30(6), 80-83.

[12] Gruenwald, L., McNutt, G. & Mercier, A. (2003), "Using An Ontology To Improve Search In A Terrorism Database System", Proceedings. 14th International Workshop on Database and Expert Systems Applications,753-757.

[13] Simons, Peter M. "Parts: A study in ontology." (1987).

[14] M. F. Lopez, A. G. Perez, and N. Juristo, "Methontology: from ontological art towards ontological engineering," in Proceedings of the AAAI97 Spring Symposium, (Stanford, USA), pp. 33–40, March 1997.

[15] M. Uschold and M. Gr¨uninger, "Ontologies: principles, methods, and applications," Knowledge Engineering Review, vol. 11, no. 2, pp. 93–155, 1996.

[16] M. Gr¨uninger and M. S. Fox, "Methodology for the design and evaluation of ontologies," in Proceedings of Workshop on Basic Ontological Issues in Knowledge Sharing held in conjunction with IJCAI-95, 1995.

[17] M. F. Lopez, A. G. Perez, and N. Juristo, "Methontology: from ontological art towards ontological engineering," in Proceedings of the AAAI97 Spring Symposium, (Stanford, USA), pp. 33–40, March 1997.

[18] V. Raskin, C. F. Hempelmann, K. E. Triezenberg, and S. Nirenburg, "Ontology in information security: a useful theoretical foundation and methodological tool," in NSPW '01: Proceedings of the 2001 workshop on New security paradigms, (New York, NY, USA), pp. 53–59, ACM, 2001.

[19] Karie, Nickson M., and Hein S. Venter. Environment." (2013).

[20] Fortis, T-F., Victor Ion Munteanu, and ViorelNegru. "Towards an ontology for cloud services." 2012 Sixth International Conference on Complex, Intelligent and Software Intensive Syst IEEE, 2012.

[21] Han, Taekgyeong, and KwangMongSim, "An ontology-enhanced cloud service discovery system" Proceedings of the International Multi Conference of Engineers and Computer Scientists. Vol. 1. 2010.

[22] Zhang, Miranda, et al. “An Ontology based System for Cloud Infrastructure Services Discovery.” arXiv preprint arXiv:1212.0156 (2012).

[23] Uchibayashi, Toshihiro, Bernady O. Apduhan, and Norio Shiratori. "A framework of an agent-based support system for IaaS service discovery." 13th International Conference on Computational Science and Its Applications (ICCSA), IEEE, 2013.

[24] Bechhofer, Sean. "OWL: Web ontology language." Encyclopedia of Database Systems. Springer US, 2009. 2008-2009.

[25] OWL guide, available at, http:// www.w3.org/TR/owl-guide/

[26] Protégé Ontology Editing Tool, available at http://protege.stanford.edu/

[27] Enron email dataset available at http://www-2.cs.cmu.edu/~enron/

[28] Protégé Wikipedia, available at, http://protegewiki.stanford.edu/wiki/Main_Page

[29] Noy, Natalya F., et al. "Creating semantic web contents with protege-2000." IEEE intelligent systems 2 (2001): 60-71.

[30] Gennari, John H., et al. "The evolution of Protégé: an environment for knowledge-based systems development." International Journal of Human-computer studies 58.1 (2003): 89-123.