Representing and Publishing Cyber Forensic Data and its Provenance Metadata: From Open to Closed Consumption

Tamer Fares Gayed, Hakim Lounis
Dépt. d'Informatique
Université du Québec à Montréal
Succursale Centre-ville, H3C 3P8,
Montréal, Canada
gayed.tamer@courrier.uqam.ca lounis.hakim@uqam.ca

Moncef Bari
Dépt. de Didactique
Université du Québec à Montréal
Succursale Centre-ville, H3C 3P8,
Montréal, Canada
bari.moncef@uqam.ca

***Abstract*—Role players of any forensic investigation process record chronologically all forensic data resulted from their investigation, in order to be presented to the juries in the court of law. When such results are recorded and posted, they are called chain of custodies (*CoCs*). The forensic data provided within these documents play a vital role in the process of forensic investigation, because they answer questions about how evidences are collected, transported, analyzed, and preserved since their seizure through their production in court. Provenance metadata accompany these forensic data to answer questions about the origin of these data and build trustworthy between role players and juries in order to make the tangible *CoCs* admissible in the court of law. Nowadays, with the advent of the digital age, the forensic investigation is not only applied to physical crime, but also on digital evidences. The forensic data and their metadata presented in these tangible documents need also to undergo a radical transformation from paper to electronic data in order to accommodate this evolution. *CoCs* should be also readable and consumable not only by human but also by machines. The semantic web is a fertile land to represent and manage the tangible *CoCs*, because it uses web principles known as Linked Data Principles (LDP), which provide useful information in Resource Description Framework (RDF) format upon Unified Resource Identifiers (URI) resolution. In addition, it includes different provenance vocabularies that can be useful to express the forensic metadata. Generally, the power of LDP resides in publishing data publicly without any access restriction on the web. However, the openness of forensic data and their metadata should not be the same case. They should obey some access restriction in order to be shared only between role players and juries. Public Key Infrastructure (PKI) can be applied to restrict the access to some or all resources of represented data and bends the LDP from open to closed consumption, while maintaining the resolution of such restricted resources. Juries in turn will consume the restricted represented data using different LDP consumption applications. This paper provides the complete framework explaining how forensic and provenance data are represented and published using LDP, and how PKI can be used to restrict these data/resources in order to be shared in a closed scale. Evaluation of the framework using several empirical experimentations will not be on the scope of this paper.**

Keywords-Linked Open Data, Linked Data Principles, Linked Closed Data, Public Key Infrastructure, Digital Certificates, Cyber Forensics, Chain of Custody.

I. INTRODUCTION

The history of forensic investigation task dates back thousands of years. This task is concentrating to gather and examine evidences about the past, in order to prosecute in the future the criminal in the court of law. With the advent of Information and Communication Technology (ICT), forensic investigation is not only concentrated on physical crime, but also on the digital evidences. This emerged a new type of forensic investigation known by computer/cyber/digital forensic. It combines computer science concepts including computer architecture, operating systems, file systems, software engineering, and computer networking, as well as legal procedures. At the most basic level, the digital forensic process has three major phases: extraction, analysis, and presentation. Extraction phase (i.e., it is also known as acquisition) saves the state of the digital source (e.g., laptop, desktop, computers, mobile phones, or any other digital devices) and creates an image by saving all digital values so it can be later analyzed [1]. Analysis phase takes the acquired data (e.g., file and directory contents and recovering deleted contents) and examines it to identify pieces of evidence, and draws conclusions based on the evidences that were found. During presentation phase, the audience is typically the judges; in this phase, the conclusion and corresponding evidence from the investigation analysis are presented to them [2][3].

However, there exist others models of cyber forensic process, each of them relies upon reaching a consensus about how to describe digital forensics and evidences [4][5]. Investigation models are numerous. Many works were provided to explain and compare such models [6][7][8][9]. Table I shows the current digital forensic models. Each row of the table presents the name of the digital forensic process model, while the columns present the processes included in each of these models [5][10].

The role players such as first responders, investigators, expert witnesses, prosecutors, police officer, etc. may be assigned one or more phase in the forensic process. They are those who are responsible to create and record their own investigation results and post them in tangible documents.

TABLE I. DIGITAL FORENSIC PROCESS MODELS [5]

	Acquire	Authenticate	Analyze	Collection	Examination	Reporting	Recognition	Identification	Individualisation	Reconstruction	Preservation	Classification	Presentation	Decision	Preparation	Approach Strategy	Returning Evidence	Awareness	Authorization	Planning	Notification	Transportation	Storage	Hypothosis	Proof/defence	Dissemination
Kruse	*	*	*																							
USDOJ			*	*	*	*																				
Casey							*			*	*	*														
DFRWS			*	*	*			*			*		*	*												
Reith			*	*	*			*			*		*		*	*	*									
Ciardhuain				*	*								*					*	*	*	*	*	*	*	*	*

These documents are known by chain of custodies, as they record all collected evidences (forensic data) in their chronological order, in order to avoid later allegations of tampering with such evidences. *CoC* considered as a testimony document and one of the most essential parts of any forensic investigation process [6], because it provides useful information about the evidences studied through different forensic process by answering 5Ws and 1H questions. The 5 Ws are the When, Who, Where, Why, What and the 1 H is the How. *CoC* must include documentation containing answers to these questions. For example:

- Who came into contact, handled, and discovered the digital evidence?
- What procedures were performed on the evidence?
- When the digital evidence is discovered, accessed, examined, or transferred?
- Where was digital evidence discovered, collected, handled, stored, and examined?
- Why the evidence was collected?
- How was the digital evidence collected, used, and stored?

Once such questions are answered for each phase in the forensic process, players will have a reliable *CoC*, which can be then admitted by the judges in the court of law.

Reliability on information is not enough to admit the *CoC* in the court of law. Trustworthiness is also required. On the level of data, it occurs when receivers (i.e., juries) ensure from the *origin* of data that the senders (i.e., role players) sent to him, and this will be realized through different provenance vocabularies. On the level of players, trustworthiness occurs when receivers ensure from the *identity* of the senders (i.e., called also repudiation), and this will be realized through the PKI. Provenance of information related to data and identities of players are crucial to guarantee the trustworthiness and confidence that the role players provided to the juries.

Thus, each *CoC* document contains not only forensic data, but also data describing the origin of this data (i.e., the forensic data presented in the *CoC* will be accompanied by metadata). Metadata is the data that describes other data. Thus, forensic information is responsible to answer the 5Ws and 1H questions related to the forensic investigation, while provenance information is responsible to answer questions about the origin of these forensic data. For example:

- Who published/created the data?
- What is the published date?
- Where this data is initially published/created?
- When/Why the data is published?
- How the data is published?

The questions of forensic data may differ from one phase to another in the forensic process. Their questions must be posed separately over each phase of the forensics process (i.e., 'What' question, of the collection phase is not the same as the 'What' for the identification phase). For example, the Kruse model (see Table I, first row) has 3 forensics phases, thus, it should have 3 different *CoCs* [11]. Nevertheless, most of works provided in the forensics process globalize the 5Ws and 1H questions once over the whole forensics process [7][9].

Nowadays, with the advent of digital age, *CoCs* should be transformed from tangible document to electronic form consumable not only by the human but also by the machine. There are three main motivations do this task [12]:

- *Motivation 1*: cyber forensics is a daily growing field that requires the accommodation on the continuous changes of digital technologies as well as its tangible documents (i.e., concurrency with the knowledge management). Thus, tangible *CoCs* and all their contents (i.e., victim information and forensics information) must also undergo a radical transformation from paper to machine-readable format in order to accommodate this continuous evolution.
- *Motivation 2*: judges' awareness and understanding the digital evidences are not enough to evaluate and take the proper decision about the digital evidence. Juries need to know more concerning the evidences in hand. One of the proposed solutions is to organize a syllabus and training program to educate the juries the field of ICT [13]. The authors argue against this solution direction, because it will not be an easy task to teach juries with their juridical positions, the different concepts of ICT. The authors propose a solution offering the ability to the juries to navigate, discover (dereference) and execute different queries on the represented information.
- *Motivation 3: CoC*s play vital role in the investigation process and due to this fact, it must be maintained and managed throughout the investigation process, in order to preserve its integrity, especially when the evidence has digital nature. However, if the *CoC* is not well maintained and the suspect was guilty, a lawyer/defense can argue that the *CoC* was not properly established and casting doubt on the damning of the acquired evidence. A security mechanism should be integrated with the represented data to keep its

integrity to limit and control its access to only the authorized people.

Semantic web will be a flexible solution for this task because it provides several semantic markup languages such as Resource Description Framework (RDF) [14], RDF Scheme (RDFS) [15], and Web Ontology Language (OWL) [16] that are used to represent different data and knowledge. In addition, the semantic web is rich with different provenance vocabularies [17], such as Dublin Core (DC) [18], Friend of a Friend (FOAF) [19], and Proof Markup Language (PML) [20] that can be used to (im)prove the *CoC* by answering the 5Ws and the 1H questions [3].

Furthermore, the semantic web today is the web of data, which is not just concentrated on the interrelation between web documents, but also between the raw data within these documents. This interrelation of data is based on three aspects known as the LDP or the technology stack. The latter contains Unified Resource Locators/Identifiers (URL/URI) [21], Hypertext Transfer Protocol (HTTP) [22], and RDF [14]. Simply, this stack is used to publish data in a structured way that facilitates their consumption through representing and naming different resources using URL/Unified Resource Identifier (URI).

The Linking Open Data (LOD) project is the most visible project using this technology stack (URLs, HTTP, and RDF) that converts existing open license and provenance data on the web into RDF according to the LDP [23][24]. Thus, the LOD publish open data on the web without access restriction. However, this will not be feasible in a context where only role players and juries need to publish and consume, respectively, the represented data in a small scale.

Represented URI/URL resources of *CoCs (e-CoCs)* need to obey then some access restriction, where a specific set of people are those who are authorized to access such resources. LDP should be bended to realize the adaptation of publishing and consuming the resources on a small scale without loosing the resolvability feature of these resources. Thus, a compromise question arises in this case, how we can realize the access restriction over certain URI/URL resources while keeping the resolvability feature of the same resources. In addition, this question brings out a new era of research called the Linked Closed Data (LCD) [25], where the publisher would take step of imposing access restrictions to protect his information [25][26][27]. Finally, the represented resources will be closed and shared only between role players and juries. The latter can consume resources using different pattern consumption; whether by browsing, crawling, querying, or reasoning on the represented data.

This paper extends the work published in [1]. In this work, a framework solution called Cyber Forensic-*CoC* (CF-*CoC*) has been provided (see Figure 1). One of the layers was the PKI layer that was used to bend the LDP from LOD to LCD. The current work resumes the work published in [1], by depicting all the layers together and by clarifying how the PKI are applied to restrict the publication and consumption of forensic data and provenance metadata.

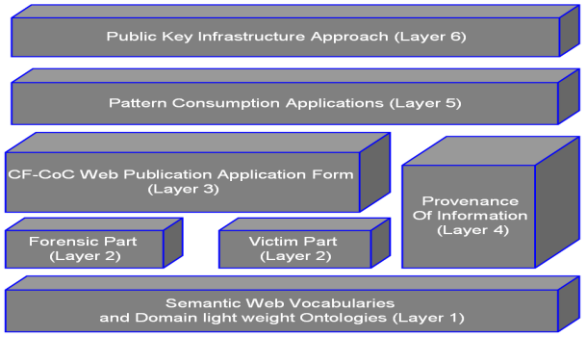

Figure 1. Cyber Forensics-Chain of Custody (CF-*CoC*) Framework

It also explains the two remaining layers of the CF-*CoC* framework (i.e., provenance metadata layer and consumption layer) that were not provided and published in our recent works [1][3][10][11][12][28].

Furthermore, as mentioned before, this work can be used to argue against the solution proposed in [13] concerning the judges' awareness and understanding of the digital evidence. This solution helps the juries to understand the field of ICT. The aim of this research is the construction of a system, offering the ability for the role player to record and publish electronically their forensic investigation and for the juries to navigate, discover (i.e., dereference), and execute different queries on the represented information in order to understand the case in hand.

This paper is organized as follows: Section II is the state of the art that depicts different disciplines related to the CF-*CoC* framework. Section III states the advantages of using LDP to represent *CoC*, Section IV provides the research problem, Section V depicts the CF-*CoC* framework and system. Finally, Section VI provides the conclusion and future works.

II. STATE OF THE ART

The state of the art related to this framework goes over different disciplines such as semantic web, cyber forensics, provenance of information, and security, so the state of the art in this section will have different facets. Each facet discusses the related works of each discipline apart (see Figure 2).

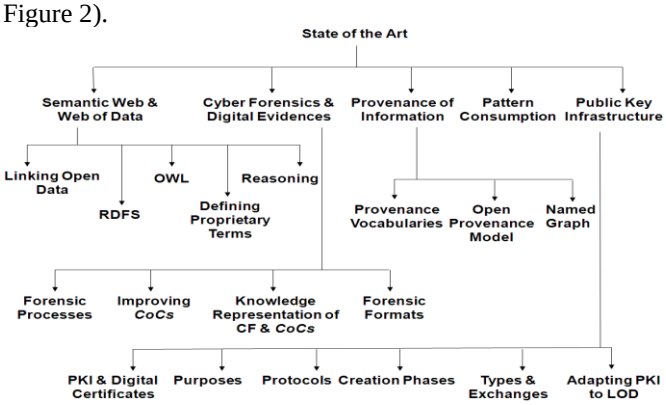

Figure 2. State of the Art related to CF-*CoC* Framework

A. Semantic Web and Web of Data

Semantic web is an extension of the current web (i.e., from document to data) [27][29], designed to represent information in a machine readable format by introducing RDF model [14] to describe the meaning of data and allows them to be shared on the web in a flexible way. The classical way for publishing documents on the web is just naming these documents using URI and hypertext links. This fact allows the consumer to navigate over the information on the web using a web browser application and crawling the information by typing keywords in a search engine that is working using the support of HTTP protocol. This is called the web of documents.

With the same analogy, entities and contents (i.e., data) within documents can be linked between each others using typed linked and with the same principles used by the web (i.e., web aspects). This is called the web of data.

Nowadays, the main aim of the semantic web is to publish data on the web in a standard structure, and manageable format [35]. Tim Berners-Lee outlined the principles of publishing data on the web. These principles known as Linked Data Principles (i.e., LD principles) [24] [27]:

- Use URI as names for things.
- Use HTTP URIs so that people can look up those names.
- When someone looks up a URI, provide useful information using the standards (RDF, SPARQL).
- Include RDF statements that link to other URIs so that they can discover related things.

According to the W3C recommendation [14], RDF is a foundation for encoding, exchange, and reuse of structured metadata. It can be serialized using different languages (e.g., RDF/XML [30], Turtle [31], RDFa [32], N-Triples [33], N3 [34]). RDF consists of three slots called triples: resource, property, and object. In addition, resources are entities retrieved from the web (e.g., persons, places, web documents, pictures, abstract concepts, etc.). RDF resources are represented by uniform resource identifiers (URIs) of which URLs are a subset.

After the resources are identified using URIs, they will be connected using RDF links, creating a global data graph that spans data sources and enable the resolvability of such resources to a new data source. The LOD cloud project has been constructed upon this basic structure.

1) Linked Open Data

The Linked Open Data (LOD) project is the most visible project using this technology stack (URLs, HTTP, and RDF) and converts existing open license data on the web into RDF according to the LDP [35][24] (see Figure 3).

The LOD is based on the LDP, where URI resources are linked using typed RDF links to other resources within the same or to other data set. Two types of links can be used; links to navigate forward and others to navigate backward between resources. For example, if we have an RDF triple connecting two resources x and y, and we need to move

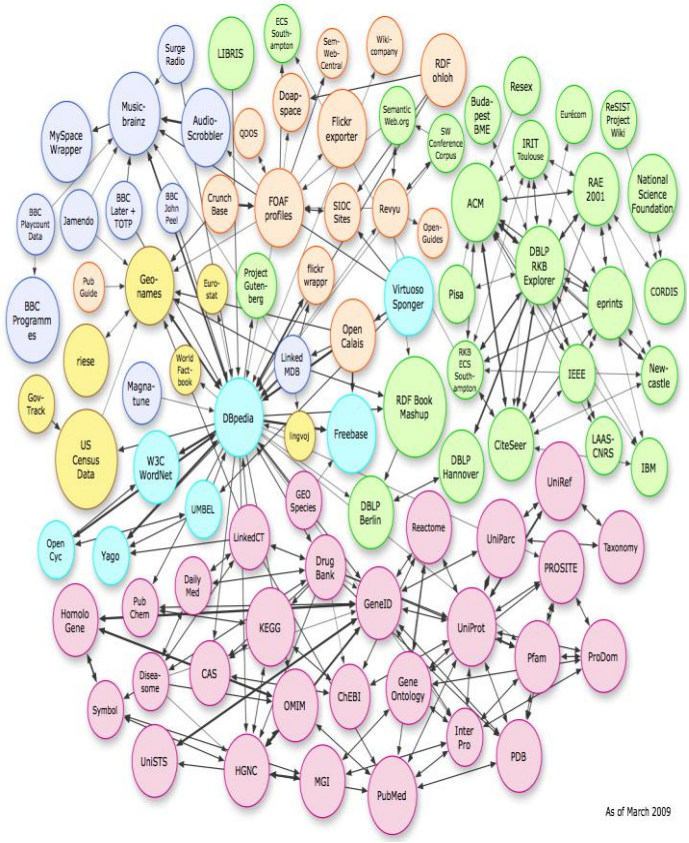

Figure 3. Linking open data cloud diagram

forward from x to y, then this RDF triple should appear in the document describing the resource y. This triple is then called incoming link, because it allows to navigate back to resource x. Same case, for the outcoming link, where the RDF triple should appear in the document describing the resource x and allows to navigate forward to resource y [36]. Figure 3 shows the LOD cloud diagram, where each links exists between items in the two connected data sets. Some data sets are connected together using whether, the outcoming links, the incoming links, or both.

The LOD project created a shift in the semantic web community. Instead, the concern was on the ontologies for their own sake and semantic, it becomes on the web aspects (how to publish and consume data on the web).

Ontologies are used then to foster and serve the semantic interoperability between parts that want to exchange such data. These are known as lightweight ontologies [37] that use the full advantages of semantic web technologies, minimum OWL constructs, and reuse existing RDF vocabularies wherever possible.

Resources have properties (attributes) that admit a certain range of values or that are attached to another resource. The object can be a literal value or a resource.

While RDF provides the model and syntax for describing resources, it does not define the meaning of those resources. That is where other technologies such as RDF Schema

(RDFS) come in [36]. RDFS specifies extensions to RDF that are used to define the common vocabularies in RDF metadata statement and enables specification of schema knowledge. It develops classes for both resources and properties. However, RDFS is limited to a subclass hierarchy and a property hierarchy with domain and range definitions of these properties. RDFS limitations are range restrictions, disability of expressing disjoitness between classes, combination between classes, cardinality restriction, and characteristics of properties [38].

Thus, RDF is the standard format to create LD and it is sufficient to use the constructors of RDFS and a little feature of OWL to represent data in LD structure. Combination of constructors from both vocabularies (i.e., RDFS and OWL) represents the lightweight ontology of RDF and LD. This is known by RDFS++. Next subsections highlight all the RDFS constructors and some OWL primitive constructors that will be used to construct the first two layers of CF-*CoC* framework.

The RDFS and OWL constructors are classified according the term type (i.e., *rdfs:class*, or a property *owl:objectProperty*). This definition takes place before the term will be used (i.e., before its publication, T-Box). Later, the defined terms are used to describe and publish different data (A-Box, Assertion Box) [38]. The type of the term also determines its slot position during publication.

2) RDFS Constructors

The RDFS constructors are used to define terms and their relationships. Consider the term in question is named *X* (see Table II).

TABLE II. RDFS CONSTRUCTORS FOR PROPERTY AND CLASS TERMS

If *X* is a term of type (*rdf : type*) Property (*rdfs : Property* / *owl : ObjectProperty*)	
rdfs : subPropertyOf	When the term *X* is of type *property* it can be also a sub property of another *property* term. The *subPropertyOf* of a property term is a term of type *Property*
rdfs : range	The *range* of a property term is always a *Class*. A *range* of a property term *X* states that the **object** slot of the *X* (i.e., where *X* is a predicate, because *X* is a *property)*, interpreted by a reasoners as an instance of said *range* of *X*
rdfs : domain	The *domain* of a property term is always a *Class*. A *domain* of a property term *X* states that the **subject** slot of the *X* (i.e., where *X* is a predicate, because *X* is a *property)*, interpreted by a reasoners as an instance of said *domain* of *X*
If *X* is a term of type (*rdf : type*) Class (*rdfs : Class*)	
rdfs : subClassOf	When the term *X* is of type *Class*, it can bee also a sub class of another *Class* term. The *subClassOf* of a property term is a term of type *Class*
Common Constructors between *Property* and *Class* terms	
rdfs : comment	Any term should have a *comment*. A *comment* is used to provide a human-readable description of a resource. *Comment* is an instance of *rdf : Property*
rdfs : label	Any term should have a *label*. A *label* is used to provide a human-readable name for a resource. *Label* is an instance of *rdf : Property*

3) OWL Constructors

The primitive selected from the OWL are mainly used to map between class and property terms (see Table III).

TABLE III. OWL CONSTRUCTORS FOR PROPERTY AND CLASS TERMS

If *X* is a term of type (*rdf : type*) Property (*rdfs : Property* / *owl : ObjectProperty*)	
owl : equivalentProperty	This constructor is used to map between two terms of type *Property*
owl : InverseProperty	This constructor is used to state that one property is the inverse of another. It is use to describe inverse relation between properties (i.e., exactly like the passive voice in the grammar)
owl : InverseFunctionalProperty	When the type (*rdf:type*) of a *property* term *X* is defined to be of *InverseFuntionalProperty*, Whenever *X property* is used as a predicate in a triple, its **object** will have **one** and **only one subject**. Thus, each object should be able to uniquely identify a subject. This constructor is a sub class of *owl : objectProperty*
owl : FunctionalProperty	Same idea as the last constructor, but here, when *X* is defined to be of type *FunctionalProperty*, each **subject,** where *X* is a predicate, can have at most one **object.** This constructor is a subclass of *rdf : property*
If *X* is a term of type (*rdf : type*) Class (*rdfs : Class*)	
owl : equivalentClass	This constructor is used to map between two terms of type *Class*
Common Constructors between *Property* and *Class* terms	
owl : sameas	Two URI terms can be mapped together using the *sameas* constructor. This constructor indicates that these two terms actually refer to the same thing. It can be used as well to map between two ontologies.

These constructors in Tables II and III, are used to publish data on web. Publication of terms on the web passes by three steps. It starts with identifying terms in the domain of interest. These terms are the things whose properties and relationships will be used later in the publication of data.

The identification (selection) is achieved through the descriptions of different processes and tasks performed within each forensic phase [39][40]. The identified terms are also called custom or proprietary terms.

Second step, the identified terms are defined using different constructors of RDFS [15] and OWL [16], and uniquely named by HTTP URIs. The defined terms are then used to publish different information.

4) Defining Proprietary terms

The existing terms defined by the vocabularies of the semantic web are not enough to describe all domains. Sometimes, there are no existing ontologies (vocabularies) containing terms describing a particular data set (e.g., cyber forensics). Some domains are new and others are still in their infancy. This is the case of CF, where it is scarce to find forensic terms or well-known vocabularies describing it, because this domain is still in its infancy and development. Thus, new proprietary terms need to be defined and developed in a dedicated vocabulary, applying the features of RDFS [15] and OWL [16] to describe this particular data set. However, before creating a new custom term, some aspects (criterions) should be taken into consideration [41]:

- Search for terms from widely used vocabularies that could be reused to describe the domain in interest. If the widely deployed vocabularies do not provide the required terms to describe such domain, so new terms should be defined as proprietary terms.
- When you define a new term, you need to have a namespace that you own and control (i.e., unique namespace), in order to mint your new terms to this domain/namespace.
- When you create new terms, you have to map these terms to those in existing vocabularies.
- Apply the LDP to your new terms by using the web technology stack (HTTP, URL, and RDF) and this

task takes place along the publication process, starting from the identification of terms until their publication.
- Label and comment each term you create.
- If your term is a property (predicate), you have to define its *domain* and *range* using the constructors of RDFS and do not overload your new term with ontological axioms.
- If at later time you discover that another term was enough, an RDF link should be set between the new created term and the existing one.

Although, there exist different guides to publish terms, the process of selecting and identifying them is remaining a subjective task and depends on the term creator (i.e., we may have two creators selecting and identifying two different terms describing the same concept in the real world). This does not affect the quality of terms being published, because the LDP on the web of data make them self-descriptiveness. The latter advantage is due to two reasons:

- LDP with naming using HTTP/URIs, offer a dereferenceable nature to the term, so that any LD consumption applications can look up the RDFS/OWL definitions and retrieve more information about such term [42].
- LDP with some schema constructors (i.e., OWL) can map a new term to existing terms from well-defined vocabularies in the form of RDF links [43].

The most related work to define an ontology in *CF*, was published in [44], where an ontological model (i.e., with small 'o') was created for outlining CF tracks in the education process. Its aim was only to construct a hierarchical structure for classification of certification domains (i.e., the best convenient vocabulary to be used by the web of data to construct such type of ontology is the Simple Knowledge Organization System – SKOS [45]). Thus, CF is a domain that requires the definition of new proprietary terms. The CF-*CoC* framework provided in this paper aids the role player to represent *CoC* by defining new proprietary terms and publish such information on the web of data in RDF format.

Any forensic process contains a set of phases, where each phase is assigned to one or more role player. Thus, the number of forensic phases and how many role players assigned to each phase determine the total number of role players participated in the forensic investigation. Each forensic phase contains a set of forensic tasks; each task can be described using a set of terms representing the forensic information.

Before discussing how the role player can use the CF-*CoC* to generate the *e-CoC* (see Section V), it is necessary to explain how the forensic information can be ontologically mapped. As shown in Figure 4, each forensic phase will have a corresponding lightweight ontology. Each lightweight ontology has a set of *n* categories, which will be equivalent to *n* forensic tasks. A category in the vocabulary should be described using a set of *m* terms. These terms are the proprietary terms describing a forensic task.

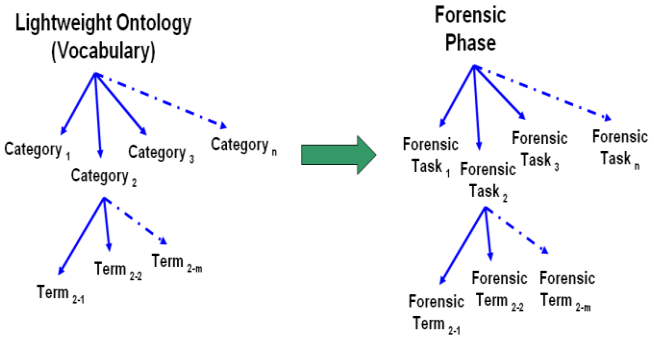

Figure 4. Mapping between Ontological Concepts and a Cyber Forensic Phase [28]

This work considered the preservation task of the acquisition phase imported from Kruse model [46], as an example to elaborate the idea of creating lightweight ontology with new proprietary forensic terms.

5) Reasoning

As mentioned in Section II.A.1, lightweight ontology of LD is a combination of RDFS constructors and some primitive of OWL. Inference is a derivation of logical conclusion from premises known or assumed to be true. Reasoning is a process to extract new information from existing information stored in a knowledge base. For the LD, the knowledge base is the RDF triples store.

RDFS and OWL contains set of inference rules related to their constructors. This section discusses the rules of RDFS constructors, and some rules of OWL (i.e., those that are primitives and used to describe the LD). Table IV depicts the rules of the most used constructors of both vocabularies (i.e., RDFS, and OWL).

TABLE IV. RULES AND ENTAILMENTS OF RDFS AND OWL [16][47]

Constructor Name	Rules and Entailments
rdfs:subClassOf	***subClassOf* is transitive when:** *(A, rdfs:subClassOf, B), (B, rdfs:subclassOf, C) => (A, rdfs:subClassOf, C)* **Another Entailment rule of *subClassOf:*** *(a, rdf:type, A),(A,rdfs:subClassOf B) => (a, rdf:type B)*
rdfs:subPropertyOf	***subPropertyOf* is transitive when:** *(a, p, b), (p, rdfs:subPropertyOf q) => (a, q, b)* **Another Entailment rule of *subPropertyOf:*** *(a, p , b), (p rdf:type rdf:Property)*
rdfs:domain	*(p, rdfs:domain, A), (a, p, x) => (a, rdf:type, A)*
rdfs:range	*(p, rdfs:range, A), (x, p, a) => (a, rdf:type A)*
owl: FunctionalProperty	*If a property p is tagged as FunctionalProperty then all x,y, and z : p(x,y) and p(x,z) => y=z*
owl: InverseFunctionalProperty	*If a property p is tagged as InverseFunctionalProperty then all x,y and z: p(y,x) and p(z,x) => y=z*
owl: inverseof	*If a property p1, is tagged as the owl:inverseof p2, then for all x and y: p1(x,y) iff p2(y,x)*

B. Cyber Forensics Processes and Digital Evidences

Second discipline in the state of the art section is related to the cyber forensic and digital evidences. Despite the infancy of the CF field, many works have been provided related to the forensic processes, *CoC*, and forensic formats.

1) First Category : Forensic Processes

The works provided under this category concentrated on the creation of different forensics processes. Different Digital Forensics Process Models (DFPM) has been proposed since 2000 (e.g., Kruse [46], the United State Department of Justice (USDOJ) [6], Casey [7], Digital Forensics Research Workshop (DFRW) [48], and Ciarhuin [8]) to assist the players of investigations process reaching conclusions upon completion of the investigation.

As been mentioned in Section I, investigation models are numerous. Many works were provided to explain and compare such models [5][6][7][8][9] (see Table I). Some phases from different forensics models may have unique technical requirement but they differ only on their names [49]. The work presented by Yussof et al. [9] underlines 46 phases from 15 selected investigation models that have been produced throughout 1995 to 2010, and then identifies the commonly shared processes between these models.

Kruse model is a model that encompasses three major phases of any forensic investigation. The three phases are acquisition of the evidence, authentication of the recovered evidence, and analysis of the evidence. The next three paragraphs explain briefly each phase apart:

- ***Acquisition:*** this phase is about acquiring digital evidences from digital suspected devices (e.g., small-scale devices, large-scale devices, etc.). It contains three forensics tasks: state preservation, recovering, and copying. The role player of this phase is the first responder [6][9].
    - *State preservation*: the first task is saving the state of the digital device under question, by seizing the machine containing the suspected device.
    - *Recovery***:** after seizing the suspected device, the role player tries to recover all deleted files on the device, especially the system files that records valuable details about this suspected device.
    - *Copy:* after recovering the deleted files, the first responder takes copy from the suspected device to avoid tampering and alteration.
- ***Authentication:*** it is the process of ensuring that the acquired evidence has not been altered and kept its integrity since the time it was extracted, to the time it was transmitted, and stored by an authorized source [39]. Any change to the evidence will render the evidence inadmissible in the court. Investigators authenticate the digital media by generating a checksum (Hash) of it contents (i.e., using the MD5, SHA, and CRC algorithms). Checksum is like an electronic fingerprint in that it is almost impossible for two digital media with different data to have the same checksums. The main aim behind this task is showing that the checksums of the seized media (suspected) and the trusted (image) are identical.
- ***Analysis:*** This is the last and most time-consuming step in this model. In this phase, the investigator tries to uncover the wrongdoing of the crime by examining the acquired data such as files and directories in order to identify pieces of evidence and determine their significance and probative value and drawing conclusion based on the evidence found. In [50], the author defined the 3 major categories of evidence that should be considered in the analysis phase:
    - Inculpatory evidence: evidence that supports a given theory
    - Exculpatory evidence: evidence that contradicts a given theory
    - Evidence of tampering: evidence that is used to tamper the system to avoid the correct identification

2) Second Category : Improving the CoCs

Several works are provided in the literature to improve the *CoC*. The work presented in [51] provides the idea of exploiting RDF structure to improve an expansible open format of AFF4. In [52], a conceptual Digital Evidence Management Framework (DEMF) was proposed to implement secure and reliable digital evidence *CoC*. This framework answered the 'who', 'what', 'why', 'when', 'where', and 'how' questions. The 'what' is answered using a fingerprint of evidences. The 'how' is answered using the hash similarity to changes control. The 'who' is answered using the biometric identification and authentication for digital signing. The 'when' is answered using the automatic and trusted time stamping. Finally, the 'where' is answered using the GPS and RFID for geo-location.

Another work in [53] discusses the integrity of *CoC* through the adaptation of hashing algorithm for signing digital evidence put into consideration identity, date, and time of access of digital evidence. The authors provided a valid time stamping provided by a secure third party to sign digital evidence in all stages of the investigation process.

Other published work to (im)prove the *CoC* is based on a hardware solution. SYPRUS Company provides the Hydra PC solution. It is a PC device that provides an entire securely protected, self-contained, and portable device (i.e., connected to the USB Port) that provides high-assurance cryptographic products to protect the confidentiality, integrity, and non-repudiation of digital evidence with highest-strength cryptographic technology [54]. This solution is considered as an indirect (im)proving of the *CoC*

as it preserves the digital evidences from modification and violation.

3) Third Category : Knowledge representation of CF Processes and CoCs

The works of the knowledge representation created in CF concentrate on the representation of the cyber forensics models or on the digital evidences (as indirect improve for the *CoCs*).

An attempt was performed to represent the knowledge discovered during the identification and analysis phase of the investigation process [55]. This attempt uses the Universal Modeling Language (UML) for representing knowledge. It is extended to a unified modeling methodology framework (UMMF) to describe and think about planning, performing, and documenting forensics tasks.

Another work provided in [5] explains how different cyber forensic processes are modeled using the UML. In this work, the behavioral Use Cases and Activity diagrams are presented in order to clarify the limitations of such processes.

A research is also provided in [56] that hypothesis that the formal representational approach will be benefit for the cyber forensics. This work summarized at a fundamental level the nature of digital evidence and digital investigation.

Other works are also presented in [57][58]. They try to improve indirectly the *CoC* through the representation of digital evidences. Both works concentrated mainly on the representation and correlation of the digital evidences and as an indirect consequence, the (im)proving of the *CoC*.

Recently, a new work is provided in [59] to model the forensic process. This work proposed an abstract model for the digital forensic based on the flow-based specification methodology. This methodology is generally used to represent several items such as data, information, or signals using the Flowthing Model (FM) that contains six stages (i.e., arrive, accepted, processed, released, created, and transferred) allowing anyone to draw the system using flow systems.

4) Fourth Category : Forensic Format

Over the last few years, different forensic formats were provided. In 2006, Digital Forensics Research Workshop (DRWS) formed a working group called Common Digital Evidence Storage Format (CDEF) working group for storing digital evidence and associated metadata [60]. CDEF surveyed the following disk image main formats: Advanced Forensics Format (AFF), Encase Expert Witness Format (EWF), Digital Evidence Bag (DEB), gfzip, ProDiscover, and SMART.

Most of these formats can store limited number of metadata, like case name, evidence name, examiner name, date, place, and hash code to assure data integrity [60]. The most commonly used formats are described here. AFF is defined by Garfinkel et al. in [61] as a disk image container, which supports storing arbitrary metadata in single archive, like sector size or device serial number. The EWF format is produced by EnCase's imaging tools. It contains checksums, a hash for verifying the integrity of the contained image, and error information describing bad sectors on the source media.

Later, Tuner's digital evidence bags (DEB) proposed a container for digital crime scene artifacts, metadata, information integrity, and access and usage audit records [62]. However, such format is limited to name/value pairs and makes no provision for attaching semantics to the name. It attempts to replicate key features of physical evidence bags, which are used for traditional evidence capture.

In 2009, Cohen et al. in [63] have observed problems to be corrected in the first version of AFF. They released the AFF4 user specific metadata functionalities. They described the use of distributed evidence management systems AFF4 based on an imaginary company that have offices in two different countries. AFF4 extends the AFF to support multiple data sources, logical evidence, and several others (im)proves such the support of forensic workflow and the storing of arbitrary metadata. Such work explained that the Resource Description Framework (RDF) [14] resources can be exploited with AFF4 in order to (im)prove the forensics process model.

Despite the multiplicity of these forensic formats, role player can use any one of them. Each forensic tool can generate one or more forensic format(s) that can describe specific forensic results (e.g., AFF4 can be generated by EnCase imaging tool, and provide information about the size of digital media, its chunk size, its chunks in segment, etc.). Role player is able to manipulate with such formats and record different information in his *CoC*. The provided framework will let the role player to define his own custom terms to describe different forensic information recorded on the *CoCs*.

C. Provenance of Information

Provenance of information is an essential ingredient of any tangible *CoC* quality. The ability to track the origin of data is a key component in building trustworthy, which is required for the admissibility of any digital evidence.

Classically, the provenance information about 'Who' created and published the data and 'How' the data is published provides the means for quality assessment. Such information can be queried and consumed to identify also the outdated information. *CoC* data source should include provenance metadata together with the forensic data. Such metadata can be exploited to give the juries data clarity about the provenance, completeness, and timeliness of the forensic information and to strength the provenance dimension for the published data.

According to the literature, different methodologies are provided by the semantic web to integrate different provenance information to the published data. Such methodologies can be classified into three main categories:

First category is using the provenance vocabularies of the semantic web [18][19][45][64]. Second category is to use Open Provenance Model (OPM) [65], and the last category is the use of Named Graph (NG) for RDF triples to add provenance metadata about each group of triples.

1) First Category : Provenance Vocabularies

A widely deployed provenance vocabularies are the Dublin Core (DC) [18], Friend of a Friend (FOAF) [19], and Simple Knowledge Organization System (SKOS) [45][64](i.e., considered as built in vocabularies on the semantic web), which contain different predicates that can provides extra information related to the published data. The objects of these predicates can be represented by URI (e.g., dereferenceable resources) or literal/terminal identifying such objects. Another provenance vocabularies provided in [17][66] describe how provenance metadata can be created and accessed on the web of data. All vocabularies presented in the semantic web can assess the quality and trustworthiness of any published data.

An example is shown in the Figure 5. In this figure, a person called Richard Cyganiak identified himself by URI http://richard.cyganiak.de/foaf.rdf#cygri and he used the *rdf:type* to specify the person class, and the FOAF vocabulary to specify his name and location. He stated that he is near to Berlin using the URI http://dbpedia.org/resource/Berlin represented in the name space dbpedia:Berlin. The latter is dereferenced and can be a subject for another RDF graph describing the City Berlin in more details: its population in which country this city is located.

Finally, the third RDF graph used the name space object of the second graph to provide what are the other cities located in Germany. All Data are expressed and enriched using different semantic web vocabularies.

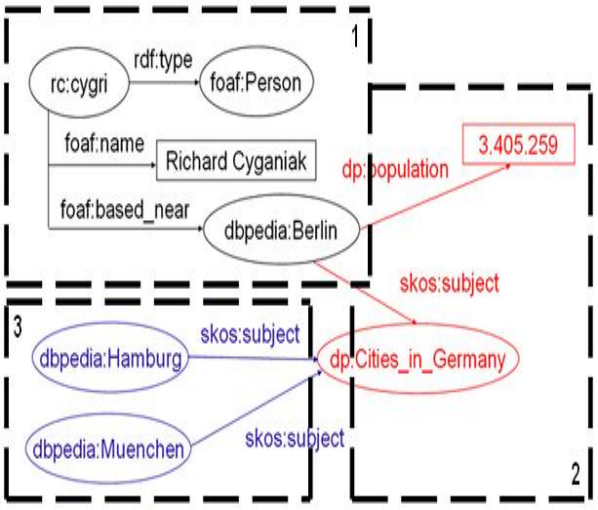

Figure 5. RDF Model with Provenance Vocabularies [67]

2) Second Category : Open Provenance Model (OPM)

Open Provenance Model (OPM) is a more expressive vocabulary that describes provenance in terms of agents, artifacts, and processes [65][68]. An extension of this work is the Open provenance model vocabulary (OPMV) provided in [65], that implements the OPM model using lightweight OWL. Open Provenance Model Vocabulary OPMV can be used also with other provenance vocabularies such as DC and FOAF.

3) Third Category : Named Graph

Whilst many authors advocate the use of semantic web technologies (i.e., vocabularies, Light weight ontologies), Caroll et al. [69] take the opposite view and proposed Named Graphs as an entity denoting a collection of triples. The idea of the named graph is to take a set of RDF triples, and considering them as one graph and assign to it a URI reference.

The NG is useful to the juries to navigate and access provenance metadata related to certain set of triples and get more description about them (e.g., LDspider [70] allows crawled data to be stored in an RDF store using the named graphs data model). As the SPARQL is widely used for querying RDF data, it can also be used in the named graph to query single or sets of named graphs. Recent work published in [23] allows publishers to add and trace provenance metadata to the elements of their datasets. This is presented through the extension of the void vocabulary into *voidp* vocabulary (i.e., lightweight provenance extension for the void vocabulary) [71]. This vocabulary considered different properties such as dataset signature, signature method, certification, and authority in order to prove the origin of a dataset and its authentication.

Simply, to illustrate how the NG can be applied in this context. If we imagine that a forensic phase (e.g., named graph of authentication, NG_{Auth}) imported from a forensic process (e.g., Kruse model) can be represented by a set of triples. Thus, the idea of the named graph will be to take this set of RDF triples (i.e., Graph) and name this graph with a URI reference.

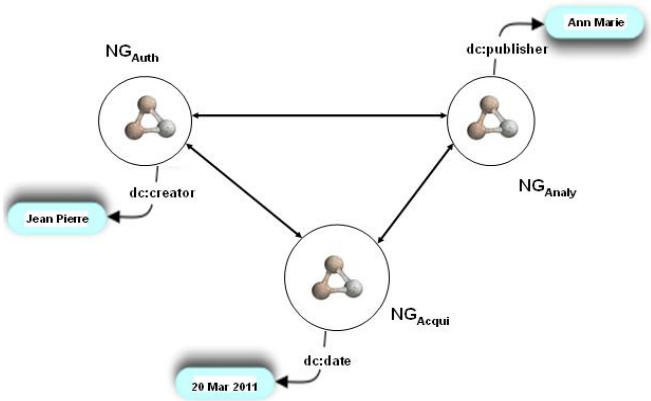

Figure 6. Named graph for Kruse Model

Figure 6 provides an abstract diagram depicting the grouping of triples and naming them to a graph with the integration of provenance metadata (e.g., DC). Each phase contains will also contain inner and outer links that relate all *CoCs* to each others.

So, the provenance metadata can be added on different levels. They can be added during the design of terms (i.e., to describe the term itself), during the publication of terms (i.e., to add more information about the data being published), or after grouping set of triples together and naming them using URI reference.

D. Public Key Infrastructure

Provenance metadata are not sufficient to ensure that the published data belong to the right players. PKI approach allows juries to ensure from the identity of role players participated in the forensics investigation.

The most related work in literature related to this paper is the one provided by Rajabi et al. in [72]. They explained theoretically how PKI is used to achieve the trustworthiness of LD and how different datasets are exchanged in a trusted way. As well, the work provided by M. Cobden et al. in [25], outlined in a vision paper, the need to have an access restriction on the LOD. Each work apart does not provide the complete picture to realize the LCD using PKI. In [72], the work explains how the PKI can be used to secure the resources of LD, but did not put the scope on how such stuffs can be implemented and applied, or how this work can bring out a new era of research related to the counter part of LOD (i.e., LCD). However, in [25], the work outlined the need of the LCD in certain domain (e.g., business and finance), but did not refer to the PKI solution, or how the LCD can be realized. Thus, this paper complements and completes the half picture of both works, by explaining how the PKI and Digital Certificates (DC) are used to restrict the access of resources in the LD cloud while keeping the resolvability of such resources, and then resulting the LCD.

This section underlines some concepts from literature related to the PKI especially the DCs; what are DCs, their purposes, their protocols, how they are created, what their types are, and how they can be exchanged. In addition, it will explain how the authors of this paper adapt PKI to LOD.

1) PKI and Digital Certificates

PKI is a combination of softwares and procedures providing a mean to create, manage, use, distribute, store, and revoke digital certificates [73][74][75][76]. PKI called Public Key because it works with a key pair: the public key and the private key

A digital certificate is a piece of information (e.g., like a passport) that provides a recognized proof of a person/entity identity. It uses the key pair managed by the PKI to exchange securely the information in order to create trustworthiness between data provider and data consumer in a network environment [77] (i.e., trustworthiness occurs when receiver ensures from the identity of the sender. As been mentioned it is known as non-repudiation).

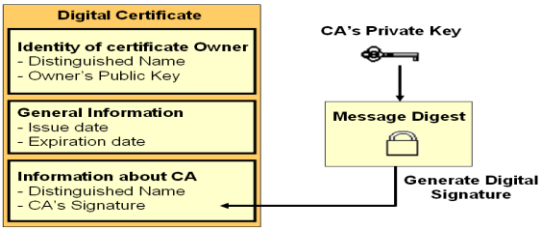

Figure 7. Digital certificate

Any certificate (see Figure 7) contains the identity of the certificate owner, such as distinguisher's name, and information about the CA (issuer of certification), such as CA's signature of that certificate, and general information about the expiration and the issue date of that certificate [78].

Digital certificate alone can never be a proof of anyone's identity. A third trusted party is needed to confirm and sign the validity and authority of each certificate and share securely the cryptographic key pair. This party is called Certification Authority (CA).

Since a CA (e.g., VeriSign Inc., Entrust Inc., Enterprise Java Bean Certificate Authority-EJBCA, etc.) relies on public trust, it will not put its reputation on the line by signing a certificate unless it is sure of its validity, the fact that makes them acceptable in the business environment.

All digital certificates provide the same level of security, whether they are created by a well-known issuer, or by unknown one. Usually, the information providers request their certificates from well-known parties when they provide services and information with large segment in society. In this paper, the authors imitate the issuer party and create CA certificate instead of buying it from well-known trusted party.

2) Purposes

A digital certificate has various security purposes and can be used to [74]:

- Allow only the authorized participant (sender/receiver) to decrypt the encrypted transmitted information (i.e., encryption).
- Verify the identity of either sender or recipient (i.e., Authentication).
- Keep the privacy of transmitted information only to the intended audience (i.e., privacy/confidentiality).
- Sign different information in order to ensure the integrity of information and confirms the identity of the signer of such information (i.e., digital signatures). Digital signatures also solve the non-repudiation problem by not allowing the sender to dispute that he was the originator of the sent message.

3) Protocols

In the field of ICT, the digital certificate is called SSL/TLS certificate because it uses two essential protocols; the SSL and the TLS [79]. The former is the short version of the secure socket layer. This protocol is used to describe a security protocol underlying a secure communication between a server and a client. After upgrading this protocol

with some encryption standards, the protocol got another acronym called TLS, which is standing for Transport Layer Security. Both protocols are based on the public key cryptography [78]. They are used to establish a secure connection over the HTTP. Classically, the HTTP establishes an unencrypted connection without using the SSL and TLS (i.e., if there is some intruder around monitoring the communication between server and client, he can come with all plain data packages of such transferred data). HTTP is then extended to HTTPS to secure the connection and encrypt all the transferred data with the SSL (i.e., HTTP + SSL/TLS = HTTPS) [80].

4) Creation Phases

The creation of a digital certificate passes by four phases (see Figure 8) using the OpenSSL tool [81]. First step, the requester (client/server/CA) generates his own pair of keys (i.e., key file), then he creates a request (i.e., req or csr format file) to the trusted party to issue for him/her a certification (i.e., crt format file). The trusted party (i.e., CA) signs the request and issues the certificate using his own private key (i.e., when the CA is the requester of the certificate, then this certificate is considered a self-signed certificate/root certificate). The created certificate is then transformed to an exportable format (i.e., p12 format) for sending it to the requester (i.e., in our context the requester is the role player).

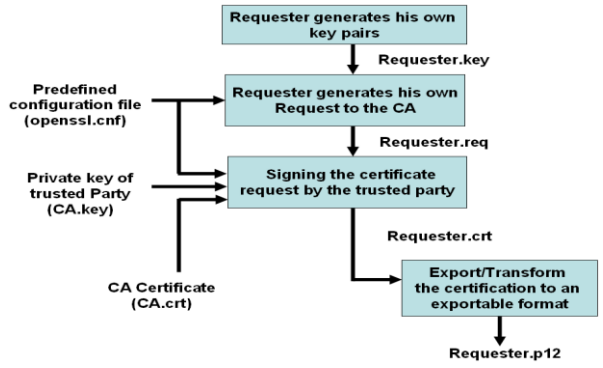

Figure 8. Procedures of creating a digital certificate using openSSL tool [1]

5) Types and Exchange

There exist three types of digital certificates. Figure 9 presents an abstract scenario where Alice and Bob want to share information over a secure connection (i.e., HTTPS).

Firstly, Alice and Bob should determine a third trusted party called the CA. The latter is responsible to issue SSL/TLS certificates for both of them in order that each can identify himself/herself to the other. CA issues two types of certificates.

- **Server certificate***:* this certificate is issued by the CA and it is used by Alice (i.e., suppose that she is the owner of the information) to identify herself to her authorized clients, like Bob. When Bob tries to access this server, he will be sure that he accessed the right server. Otherwise, Bob will not trust Alice information.
- **Client certificate**: the CA issues this certificate, and it is used by Bob (i.e., suppose he is the consumer of Alice' information) to identify himself to Alice. Alice will not allow any one to access her information unless he has a certificate known by her.
- **CA certificate:** CA also has the own certificate to sign the certificate requests received from the clients and servers. In addition, this type of certificate answers the question of how Alice and Bob ensure the identities of each others. Alice would know that Bob is the right person by verifying that his certificate is signed by the common trusted part authority (CA), as well as for Bob. Both know each others through the CA certificates.

From the definitions mentioned above, we notice that there is no distinguishable difference between the server certificate and the client certificate; both use the certificates to identify themselves to each other. However, the only difference that distinguishes both is about who is providing the information and who will go to consume it.

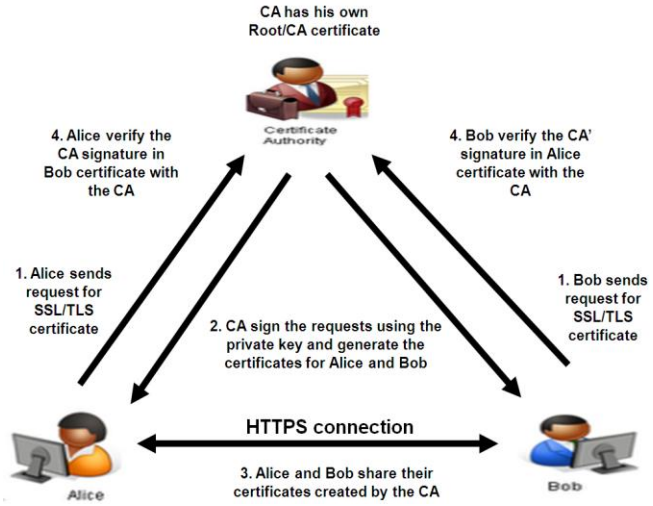

Figure 9. Sharing SSL/TLS certificates [1]

6) Adapting PKI to LOD

In this sub-section, we will discuss how digital certificates can be applied to LOD to publish and consume data on a small scale. In other words, this section describes how digital certificates are used to restrict the access of certain resources and at the same time, such resources will be resolvable to more resources.

Referring to Figure 3 of the linking open data cloud diagram, we find several data sets interrelating using outer and/or inner links. Each data set is published in a unique domain owned only by the publisher of this data set over the WWW space. Each data set contains set of URI resources that are interrelated between each others within the same data set or to an outer data set.

Now, imagine that the owner of a data set wants to publish resources using the technology stack/LDP of the LD (URI, HTTP, and RDF) and having such resources resolvable within the LOD cloud, but at the same time, he wishes to publish them in a manner that any anonymous parties on the web space cannot access them.

The idea to realize both features at the same time (i.e., resolvability and access restrictions of resources) resides in the digital certificates. The latter can be used to restrict the resolvability of resources in a one-way manner. With other words, the resources are restricted using digital certificates to be forward resolvable, but not backward resolvable unless the owner of such resources specify and list his authorized clients existing outer of his domain to access his resources. Same concepts can be applied between data sets/resources in the LOD cloud, where each data set owns a digital certificate(s). Thus, publisher of the resources can accomplish his publication task through an enhanced technology stack using a secure access protocol (i.e., HTTPS). Therefore, the current technology stack is transformed from (URI, HTTP, and RDF) to (URI, HTTPS, and RDF).

Imagining a scenario will be as follow: assuming that the publisher (server) and consumer (client) of the LD have already a common trusted party to issue their certificates. The publisher has a domain name named by an IP (i.e., for simplicity consider this IP is corresponding to a domain string name in [82]) to publish his resources in the LOD cloud. The publisher of this domain wants only someone called: 'Jean-Pierre' to consume his resources from his domain within the LOD cloud. In this case, the publisher of the data has restricted the access to his resources to a specific consumer, but he is still able to dereference his resources and resolve them to retrieve more resources outside his dataset/domain. Publisher will be also able to move back to his domain using the backward link, because he owns the server certificate for this domain. Any other anonymous party outside this domain will not be able to access the resources of [82]. If the publisher wants someone else rather than 'Jean-Pierre' accesses his resources, this person should have a client certificate signed by the same trusted party.

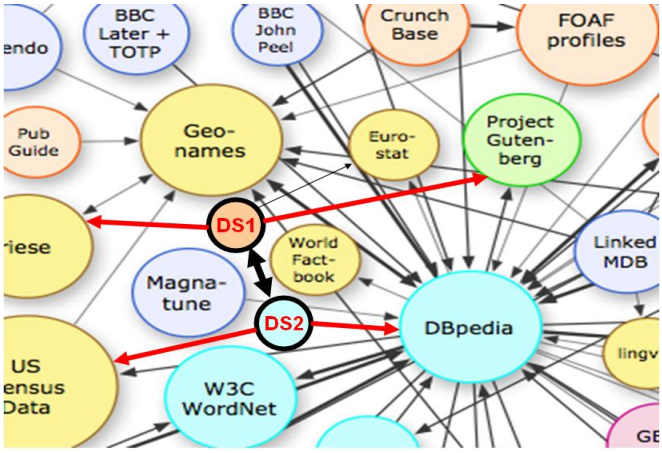

Figure 10. Client/Server certificate between two data sets

Talking in a LD manner, we can not only consider the client side as a person (i.e., as Jean-Pierre to access restricted resources), but the client side can also be a dataset or a resource within a data set that can access other resources in another data set using outer links (i.e., by moving backward to the publisher resources). In addition, another important point should be underlined; Jean-Pierre/dataset/resources can react also as a server side, if we look to the picture from the inverse direction.

Thus, Jean-Pierre/dataset/resource may have also a server certificate for his/its domain and allows the access to only people/dataset/resource that has a client certificate to his/its domain.

To illustrate this idea, Figure 3 of the LOD cloud is zoomed-in, resulting in Figure 10. Let us consider that we have two data sets DS1 and DS2 residing in two different domains. Each domain represents a data set. Both of them are interrelated between each others using inner and outer links. As well, both data sets are related with other data sets in the LOD cloud.

To elaborate the idea in terms of dataset, let consider the DS1 and DS2 can be client and server at the same time. If we look from the DS1 to DS2, we will see an outer link from DS1 to DS2 and vice-versa. DS1 is considered as a client trying to access the server DS2. Thus, DS1 will have a client certificate for its domain to identify itself to the server certificate installed in the DS2 domain. Now, let us consider if we have the contrary view; DS2 should has then a client certificate to access the server DS1 resources. However, for any other data sets around the scope of DS1 and DS2, they will not be able to resolve their resources with resources from DS1 and DS2 (i.e., at this time, DS1 and DS2 act as servers and requires client certificates from their surrounded data sets). Therefore, the resources of DS1 and DS2 have access restriction while their resources are resolvable with different resources from the LOD cloud, but the latter cannot resolve their resources from the two data sets, DS1 and DS2.

Furthermore, the certificates cannot only used on the level of datasets (i.e., including all resources), but can also be issued on the level of a specific resource within the datasets. This can be realized by issuing the certificate using one of the three URI patterns provided in Section II.

E. Pattern Consumption Applications

LD is a style of publishing data that makes it easy to interlink, discover, and consume them on the semantic web. The first way to publish LD on the web is to make URIs that identifies data items dereferenceable into RDF descriptions. Consumers can use three different patterns (i.e., in this context it will be the juries) to consume the information (i.e., the *CoC* published by role players): browsing, searching, and querying. Browsing is like the traditional web browsers that allow users to navigate between HTML pages. Same idea applied for LD, but the browsing is performed through the navigation over different resources by following RDF links and downloads them from a separate URL (e.g., RDF browsers such as Disco, Tabulator,

or OpenLink Browser). A custom semantic specialized for the juries can be easily created [83].

RDF Crawlers are developed to crawl LD from the web by following RDF links. Crawling linked LD is a search using a keyword related to the item in which juries are interested (e.g., SWSE [84] and Swoogle [85]). Juries can also perform extra search filtering using query agents. This type of searching is performed when SPARQL endpoints are installed, allowing expressive queries to be asked against the dataset. Furthermore, a void vocabulary (vocabulary of interlinked datasets) [71] contains a set of instructions that enables the discovery and usage of linked datasets through dereferenceable HTTP URIs (navigation) or SPARQL endpoints (searching), using SPARQL protocol (*void:sparqlendpoint*) or URI protocol.

In this work, we will present a framework to solve different problems related to these facets. Nowadays, the tangible CoC presenting the digital evidences and their forensic information, need to undergo a radical transformation from paper to electronic data readable, discoverable, understandable, and consumable by people and computers. This transformation helps to accommodate the evolution of digital technologies. In addition, the nature of the cyber forensic field needs a solution that unifies and represents forensic information and its formats [63] in a unified framework [14] (i.e., first and second facets). In addition, the forensic information that we want to represent needs to be shared securely on a small scale between only the role players and juries. This fact necessitates the usage of a security algorithm (i.e., PKI approach, fourth facet). Furthermore, the security of information is not enough to build trustworthiness between both parts (i.e., publisher of information represented by role players, and consumer of this information represented by juries). The admissibility of the represented information is also mandatory. The ability to track the origin of data is a key component in building trustworthy of information in order to be admissible and accepted in the court of law. Thus, Provenance of information is also required (i.e., third facet). Finally, the forensic and its provenance information needed to be available and consumed by juries through different patterns consumption applications. The latter will help them to understand and take the proper decision towards the represented information. These problems will be explained also in details in Section IV.

III. ADVANTAGES OF USING LDP FOR REPRESENTING COC

This section depicts explicitly all the advantages of using the LDP to represent the *CoC*. Knowledge *representation* has been persistent at the centre of the field of Artificial Intelligence (AI) since its founding conference in the mid 50's. Davis et al. [86] describe this concept through five distinct roles. The most important is the definition of knowledge representation as a surrogate for things. Thus, before explaining how each layer in the solution framework works, the authors decided to underline why LD is selected to represent the tangible *CoC* in cyber forensics. Thus, this section lists all the advantages and the common features of using LD to represent the *CoC* for cyber forensics:

1. *CoC* and LDP are metaphors for each others. The nature of *CoC* is characterized by interrelation/dependency of information between different phases of the forensics process. Each phase can lead to another one. This interrelation fact is the basic idea over which the LD is published, discoverable, and significantly navigated using RDF links. RDF links in LDP will not be used only to relate the different forensic phase together, but it can also assert connection between the entities described in each forensic phase. In addition, RDF typed links enable the data publisher (role player) to state explicitly the nature of connection between different entities in different and also same phases, which is not the case with the un-typed hyperlinks used in HTML.
2. LD enables links to be set between items/entities in different data sources using common data model (RDF) and web standards (HTTP, URI, and URL). As well, if the *CoC* is represented using the LDP, the items/entities in different phases can be also linked together in forensics process. This will generate a space over which different generic applications can be implemented:
   - *Browsing applications*: enable juries to view data from one phase and then follow RDF links within the data to other phases in the forensics process.
   - *Search engines*: juries can crawl the different phases of the forensics process and provide sophisticated queries.
3. LD applications that are planned to be used by juries will be able to translate any data even it is represented with unknown vocabulary. This can be realized using two methodologies. First, by making the URIs that identify vocabulary terms dereferenceable (i.e., it means that HTTP clients can look up the URI using the HTTP protocol and retrieve a description of the resource that is identified by the URI) so that the client applications can look up the terms, which are defined using RDFS and OWL. Secondly, publishing mappings between terms from different vocabularies in the form of RDF links. Therefore, for any new term definition, the consumption applications are able to provide and retrieve for the juries extra information describing the provided data.
4. Nowadays, RDFS [15] and OWL [38] are partially adopted on the web of data. Both are used to provide vocabularies for describing conceptual models in terms of classes and their properties (definition of proprietary terms). RDFS vocabularies consist of class *rdfs:class* and property *rdf:property* definitions, which allow the subsumption relationships between terms. This option is useful for juries to infer more information from the data in hand using different reasoning engines. For

example, RDFS uses a set of relational primitives (e.g., *rdfs:subclassof*, *rdfs:subpropertyof*, *rdfs:domain*, and *rdfs:range* that can be used to define rules that allow additional information to be inferred from RDF graphs). Also, OWL extends the expressivity of RDFS with additional modeling primitives that provide mapping between property terms and class terms, at the level of equivalency or inversion (e.g., *owl: equivalentProperty*, *owl:equivalentClass*, *owl:inverseof*).

RDFS and OWL are not yet fully adopted on LDP, but soon the full adaptation will be achieved [87][88][89]. This will be a great advantage to add more property and class terms to the semantic dimension of the LD, and therefore, provide useful and descriptive information.

5. Representing *CoC* data using LDP will be enriched with different vocabularies such as Dublin Core (DC) [18], Friend of a Friend (FOAF) [19], and Semantic Web Publishing (SWP). In addition, vocabulary links is one type of RDF links that can be used to point from data to the definitions of the vocabulary terms, which are used to represent the data, as well as from these definitions of related terms into other vocabularies. This mixture is called schema in the LD; it is a mixture of distinct terms from different vocabularies to publish the data in question. This mixture may include terms from widely used vocabularies as well as proprietary terms. Thus, we can have several vocabulary terms to represent the forensics data and make it self descriptive (i.e., using the two methodologies mentioned in point 3) and enable LD applications to integrate the data across vocabularies and enrich the data being published.
6. Juries need to avoid heterogeneity and contradictions about the information, which are provided to them in the court in order to take the proper decision. LD tries to avoid heterogeneity by advocating the reuse of terms from widely deployed vocabularies (same agreement of ontology). LDP is then useful to represent this type of information.
7. As mentioned in point 1, a forensics process contains several phases, which are dependent and related to each others. Each entity is identified by a URI namespace to which it belongs. An entity appearing in a phase may be the same entity in another phase. The result is multiple URIs identifying the same entity. These URIs are called URI aliases. In this case, LD rely on setting RDF links between URI aliases using the *owl#sameas* that connect these URIs to refer to the same entity. The advantages of this option in *CoC* representation are:
   - *Social function*: investigation process is a common task between different players. The descriptions of the same resource provided by different players allow different views and opinions to be expressed.
   - *Traceability*: using different URIs for the same entity allows juries that use the *CoC* published data to know what a particular player in the investigation process has to say about a specific entity of the case in hand.

   Same thing occurs not only at the level of URI but also at the level of terms. Players of the forensics process may discover at a later point that a property vocabulary contains the same term as the built in one. Players could relate both terms, stating that both terms actually refer to the same concept using the OWL (*owl: equivalentClass*, *owl: equivalentProperty)* and RDFS vocabularies (*rdfs: subclassOf*, *rdfs:subProperyOf*).
8. Provenance metadata can also be published and consumed on the web of data [66]. Such metadata provide also an answer to six questions, but at the level of the data origin (i.e., see Section I for provenance questions). These vocabularies can be used concurrently with the forensics data, to describe their provenance and complement the missing answers related to the forensic investigation.

IV. Research Problems And Methodologies

As been mentioned, the *CoC* is a testimony document that records all information related to the evidences (digital/physical) in order to ensure that they are not altered throughout the forensic investigation. Failure to record enough information related to the evidence may lead to its exclusion from the legal proceedings.

Nevertheless, the existences of many works for the *CoC* in CF there are still several issues preventing the role players to securely record, describe, and manage the results of their forensic investigation. In addition, these problems complicate the task for juries to consume and understand the digital evidences and take the proper decision about the provided information. Some problems may be resolved in the literature using another way or using classical (non-technological) methods. This section will summarize explicitly how the novel CF-*CoC* framework uses new existing technologies to solve such problems.

A. First problem : Accomodation with technology evolution

As mentioned in Sections I and II, cyber forensic is a daily growing field that requires the accommodation on the continuous changes of digital technologies (i.e., concurrency with the knowledge management). Each forensic process is associated a tangible *CoC* document that need to undergo a radical transformation from paper to machine-readable format to accommodate this continuous evolution. The LD has widely established de-facto standards (RDF, SPARQL) for sharing and interlinking of data on the semantic web.

In addition, the forensic information resulted from the forensic tools need to be interoperable with the represented *CoC*s in order to obtain a complete picture about the accomplished investigation process. AFF4 [61][63] is an open format for the storage and processing of digital evidence. Its design adopts a scheme of globally unique identifiers (URN) for identifying and referring to all

evidence [63]. The great advantage of this format is representing different forensic metadata in the form of RDF triple (subject, predicate, and value). The subject is the URN of the object the statement is made about, and the predicate (e.g., datelogin, datelogout, evidenceid, affiliation, etc.) can be any arbitrary attribute, which can be used to store any object in the AFF4 universe. Representing such formats in the same unified framework (RDF), facilitates as well the consumption of all information resulted from the forensic tools.

The CF-*CoC* will use the semantic web as a fertile land to create interlinking *e-CoCs* readable and consumable by the machine and the forensic information resulted from a forensic tool can be interoperable with these interlinking *e-CoCs (layer 1,2,3, and 5)*

B. *Second problem : security of represented CoC*

Second problem concerns the security of the *CoC* documents. Usually, the *CoC* documents must be affixed securely when they are transported from one place to another. This is achieved using a very classical way: seal them in plastic bags (i.e., together with physical evidence if there is any, such as hard disk, USB, cables, etc.), label them, and sign them into a locked evidence room with the evidences themselves to ensure their integrity. The *e-CoCs* need also to be secured since their publication by the role players until their consumption by the juries. LDP are used to publicly publish the data on the web and need to be adapted with some access and license restrictions.

The CF-*CoC* will use PKI to securely publish and consume the data in a small scale between the role players and juries *(layer 6)*.

C. *Third problem : Build trustworthiness between role players and juries*

The problem is not only to represent the knowledge of the tangible *CoC* in order to solve the issues mentioned above, but also to express information about where the *CoC* information came from. Juries can find the answers to their questions on the *CoC*, but they need also to know the provenance and origins of those answers. As been mentioned, provenance of information is crucial to guarantee the trustworthiness and confidence of the information provided. Provenance information is responsible to answer questions about the origin of answers (i.e., what information sources were used, when they were updated, how reliable the source was).

The CF-*CoC* will use provenance metadata imported from different vocabularies of the semantic web. Such vocabularies can be useful to answer the questions about the origin of the *CoC* data. Providing answers to such questions make the *e-CoC* admissible to the court of law *(layer 4)*.

V. CF-*CoC* Framework And System

The CF-*CoC* framework presented in Figure 1 is constituted of several layers. Each layer is responsible to perform certain task. The order in which the layers are placed is just to provide a conceptual diagram and explains the different tasks needed to convert a tangible *CoC* into electronic data. The number assigned to each layer, it is just for numeration. For example, the PKI layer is numerated by number six, and it is placed as last layer. This does not mean that it will be used as a last task. However, it can be used along several tasks (i.e., before defining terms, during publishing of terms, or during consumption). It is just placed at the top to globalize that it can be applied to any of their antecedent layers. Another example is the provenance of information layer, this layer can be applied to the term being designed, or to the terms being published. As we mentioned in Section II: the provenance of information may be used to describe the terms during their design, during their publication, or after publishing set of triples describing certain forensic phase.

Thus, the provenance layer describes the addition of different provenance metadata to the forensic information being published. Juries can then query and consume this information and its metadata from the consumption layer. The latter provides different consumption applications to the juries in the court of law.

This section describes in details how each layer can be built and implemented. Different modules are implemented in the CF-*CoC* system (see Figure 11), and each module contains different tasks.

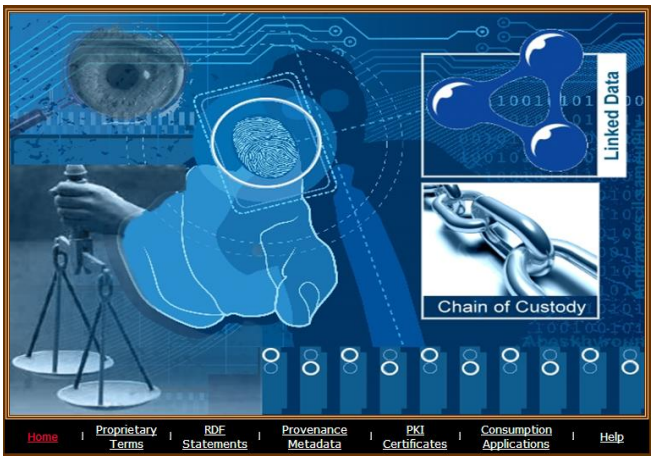

Figure 11. CF-*CoC* System

First module dedicated to create proprietary terms. This module is used to identify and design new custom terms to describe forensic information. A module for creating set of RDF statements. This module is used to publish different RDF triples by using different proprietary terms (i.e., custom terms created by publisher/role players) and build in terms (i.e., terms defined by well-known vocabularies of the semantic web). Different provenance metadata can be added along the tasks of both modules using the provenance metadata module. A PKI certificates module is also integrated on the system to create different types of

certificates to bend the designed and published terms and triples on a small scale to be shared only between juries and role players. Finally, juries consume such represented information using the consumption module.

A. Work Environment

The CF-*CoC* framework is implemented using Php and easyRDF [90], Graphiz tool [91], and its graph objects are used within the easyRDF to produce and draw different RDF models. In addition, the operating system used in this experimentation is Windows XP, accompanying with the Internet Information Services (IIS) [92] and the Openssl tool [93]. IIS simulate the machine as a server, and the OpenSSL tool is used to create the digital certificates.

B. CF-CoC Terms Definitions (Layer 1 and 2)

As shown in Figure 4, the first step is to create the ontology corresponding to a forensic phase. This ontology will contain all the forensic terms describing the different tasks of acquisition phase. In the tasks of creating ontologies, proprietary terms, and publication of data using such terms, we assume that the publishers (i.e., role players of a forensic process) own background knowledge of how to create ontologies and publish RDF data. Each role player will define his own terms from his point of view.

1) Creation of Ontology (Vocabulary):

The main objective of creating ontology objects is to create proprietary terms. Ontology object in the LD acts as a container for creating custom terms. The role players are responsible to create such objects. In LD, it is sufficient to create the ontology object and add provenance information to it (i.e., publisher name, date of creation, label, and comments). After creating ontologies, the role player appends and creates proprietary terms to his ontology(ies). Other role players can share these custom terms to publish their own data. Thus, ontologies can be reused and shared by any role player, and each role player has the liberty to use an existing ontology describing certain forensic task, or create his new ontology to describe the same forensic task. In LD, there is no negative effect to create more than one ontology describing the same task, because by creating more ontologies, we have to define more terms describing these ontologies (i.e., corresponding to forensic phase). If there exist any redundant term, a mapping process can be performed to align such ontologies. Therefore, in the LD creating ontologies is an intellectual and subjective task not as the semantic web to create full and detailed common ontologies. By time, system will contain different ontologies describing different forensic phases, created by different role players that have different point of views.

The task of creating ontologies is about to create the ontology object of the acquisition phase (see Figure 4). The domain name field is required to mint the ontology to a unique domain name owned by the publisher (aspect 2). The second field is about the selection of role player certificate [1]. In addition, the value type of the role player can be a resource or a literal. Next fields are the ontology name and its label description.

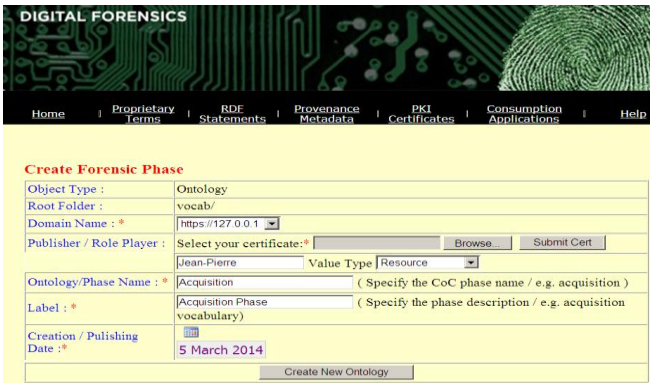

Figure 12. Creation of Acquisition Ontology

Last field is the publication date of the acquisition ontology (see Figure 12).

After completing this form, the acquisition ontology is generated by using the *Graphiz* module [91] (see Figure 13).

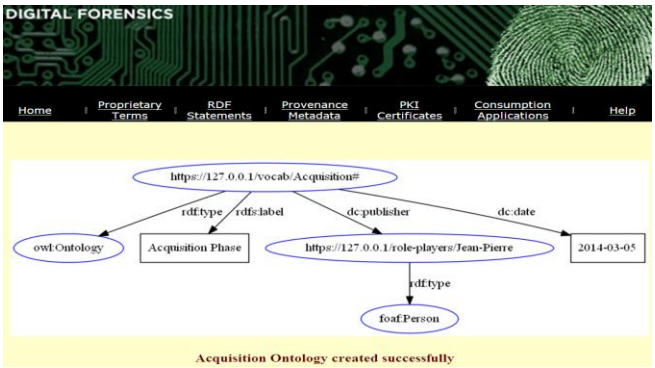

Figure 13. Creation of Acquisition Ontology

After creating the acquisition ontology, the role player proceeds with the next module to create terms and append them to this new ontology object.

2) Creation of new terms:

This task relates to four essential fields. The first field is the term name. The second field is selecting ontology to append the new proprietary term. The third field specifies the category/forensic task (see Figure 14). In our case, the category could be one of the three tasks provided in Section II (preservation, recovery, or copy). In this field, the user may select 'New' to create a new category or select 'Existing' to import an existing category, defined in another vocabulary (ontology) created by another role player (i.e., two different forensic phase may have a common category/task). Last field is the selection of term type (i.e., a term can be a property or a class).

As an example, consider the following tangible *CoC*:

"The name of the first responder in the acquisition phase is Jean-Pierre. He is the role player of this phase, and he preserved the state of the digital media, PDA device, which has the SN: 0G-4023-32-362. The date he did this task is 5 March 2014".

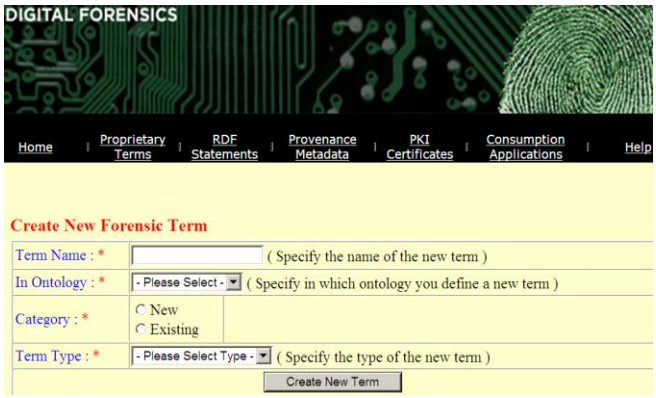

Figure 14. Creation of a New Term

The first step to create an *e-CoC* from this tangible *CoC* is to identify the terms (see Table V) (i.e., as we mentioned in Section II.A.4, identifying proprietary terms are subjective task and may differ from one creator to another).

TABLE V. PROPRIETARY TERMS OF PRESERVATION TASK

	Term name	Type
T- Box	First_responder	Class
	Role_player	Class
	Acquisition	Ontology
	Digital_media	Class
	preserve	Property
	preservedby	Property
	SN	Property
A- Box	Jean-Pierre	Subject/Object
	PDA-device	Subject/Object
	0G-4023-32-362	Object

This case study contains T-Box and A-Box information. Terms of T-Box are of type class and property. The *Role_player* term is a class that can be defined as a subclass from the class *Person* in the *FOAF* (friend of a friend) ontology (see Figure 15) [16][19]. This term will belong to a forensic task called *Preservation* task.

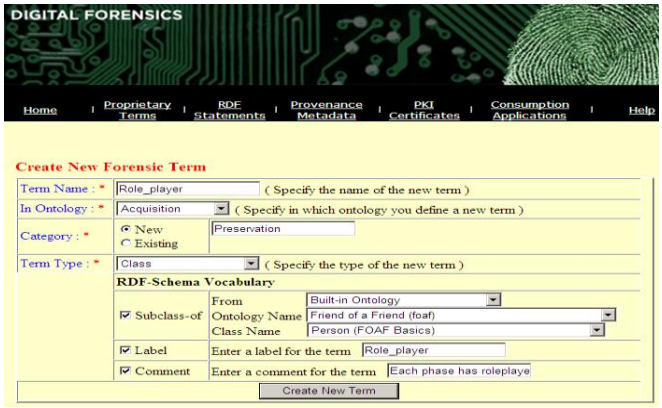

Figure 15. Creation of the Role_player Class

The *First_responder* term is a class that can be an instance of the *Role_player* class. Now, the *Preservation* category will be found under the 'Existing' category. Finally, the *Digital_media* is a subclass of *owl:Thing* (see Figure 16).

Now, the property terms (*owl:objectProperty*) will be defined. The *domain* and *range* of the term *preservedby* are defined to be *Digital_media* and *First_responder* class, respectively. This property term is defined to be a sub-property from *foaf:made* property (see Figure 17).

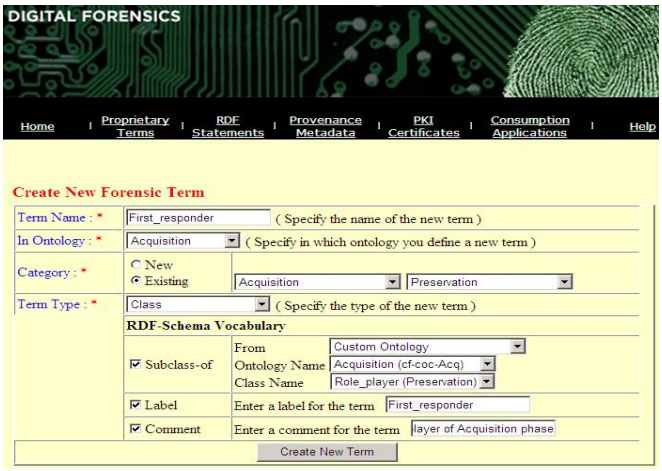

Figure 16. Creation of the First_responder Class

The *preserve* property is the inverse of *preservedby* property. Thus, the *domain* and *range* of the former will be also the inverse, *First_responder* and *Digital_media* respectively. Simply, if a digital media is preserved by a first responder, then this means that the first responder preserved the digital media. The last property is *SN*: the serial number of a device is an inverse functional property, because each serial number identifies one and only one subject (see Table III).

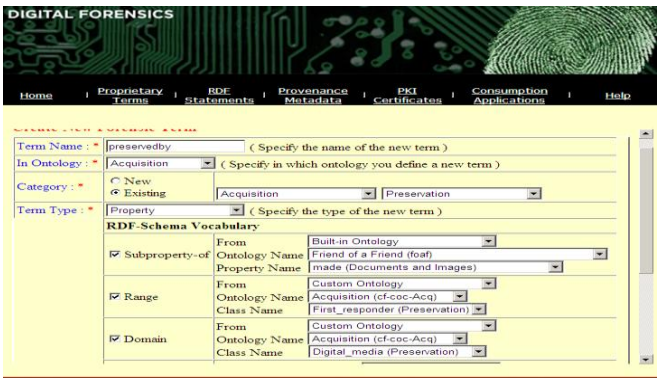

Figure 17. Creation of the preservedby Property

After creating all terms, the role player can generate the acquisition ontology with all the property and class terms of the preservation forensic task (Figure 18).

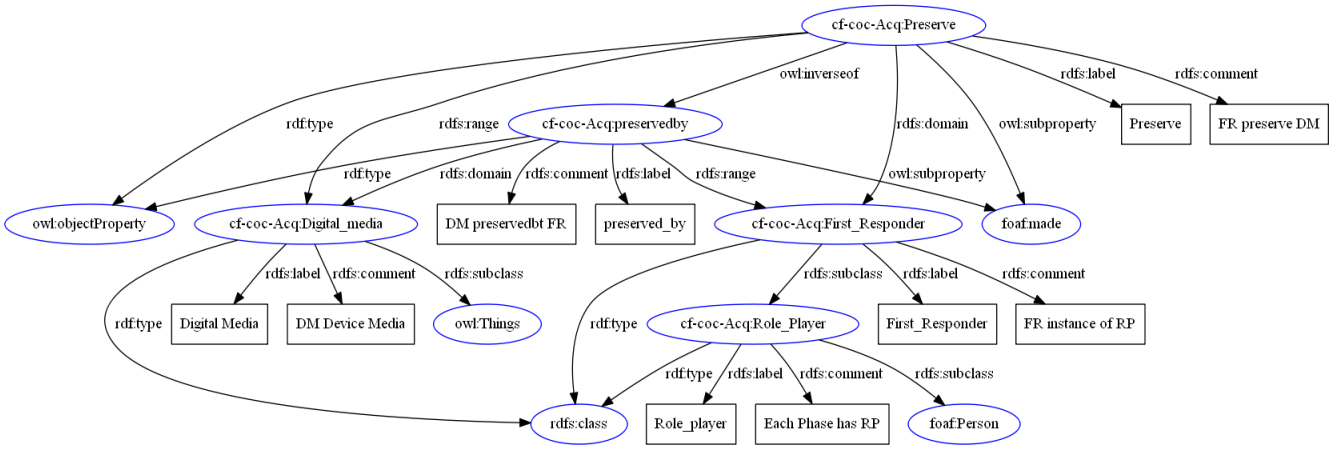

Figure 18. T-Box Ontology of Forensic Preservation Task

The T-Box terms created will be used to publish data. Therefore, they will describe the A-Box data. The latter is the *e-CoC* that will be consumed later by jury in the court using different pattern consumption (browsing [94] [95], querying [96], or searching [97]). Now, the data can be described and published using the terms defined in the T-Box and by using the third layer of the framework, the user can publish different triples (i.e., using different vocabularies of the semantic web) and by the support of proprietary vocabularies defined by the role player.

C. Publications of RDF Statements (Layer 3)

This layer is straightforward. All custom terms that have been defined in proprietary terms module (T-Box) can be used to publish and describe the *CoC* in form of RDF triples. Not only custom terms are used to publish RDF statements, but also the terms from the well know vocabularies can be used to publish such RDF statements.

The main tasks in this module are the publication of terms and mapping between them. Publication of terms is about selecting the subject, predicate (property), and object. For mapping between terms, different constructors from OWL vocabulary can be used such as *equivalentProperty*, *equivalentClass*, and *sameas* (see Table III).

The main axis over which the RDF statement is constructed is the property slot of the triple. This slot is essential to publish any RDF statement, because on its left (subject) the domain of term is defined and on its right (object) the range of the term is defined (see Table II), and then in turn, the classes and subclasses are defined. Property term (predicate) is considered as the initial node of any T-Box. From this starting node, all leaves (non-terminal) are expanded until reach the literals are reached (terminal leaves).

For example, the property term 'preserve' defined in the T-Box, its domain is First_responder class (Subject) and its range is Digital_media class (object). Thus, any literal given by the publisher in the subject slot of RDF triple will be of type First_responder, which is a subclass of Role_Player, which is a subclass of class Person (i.e., defined in the foaf vocabulary); see Figure 19.

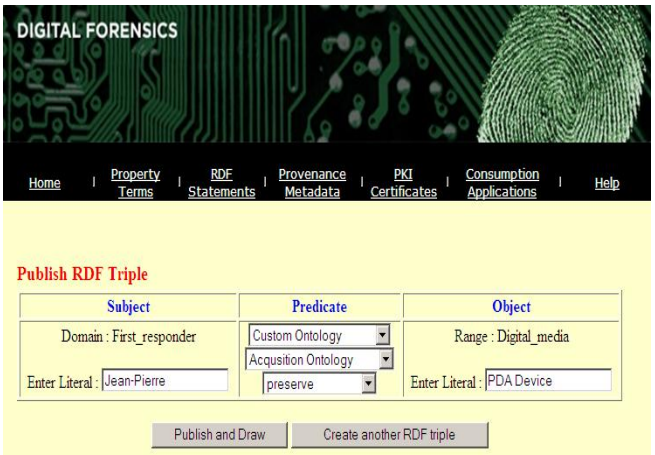

Figure 19. Publish RDF Triple

The second main task of this module is mapping between terms. The predicate slot of this triple will be one of three constructors mentioned above (see Table III), the subject and object are terms of type class or property, in order to map term class to term class or term property to term property. An example to map two different terms are those of type First_responder and Role_Player. In some cases, the role player is the generic term used to any forensic process, and in other cases the exact player is identified by its role (i.e., the role player is assigned to the acquisition phase and at the same time the player of this phase is called the First_responder). Thus, a term of type First_responder can be equivalentClass to another term of type Role_Player. Describing a term with different point of

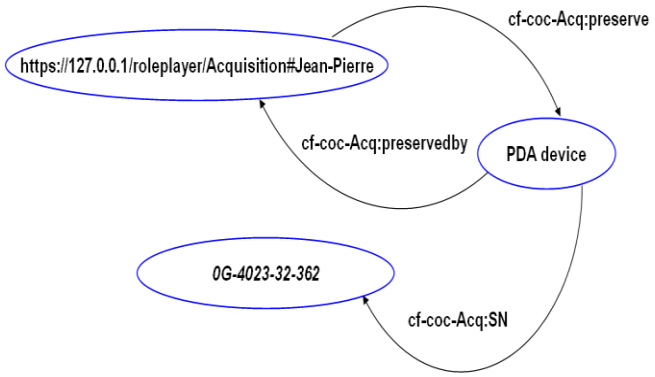

Figure 20. A-Box Ontology of Forensic Preservation Task (*e-CoC*)

views ameliorates the result of any reasoning engine (e.g., primer) [98].

Figure 20 shows the *e-CoC* (A-Box) of the forensic preservation task. This generated ontology does not answer all the question of *CoC*. It answers only the Who: Jean-Pierre, What: PDA device, and When: publication date of ontology. In order to have the answers to other questions, more terms need to be determined and defined. In this figure, the *cf-coc-Acq* is the prefix namespace of the acquisition ontology: *Jean-Pierre* is an instance from the *First_responder* class (i.e., which is an instance of the *Role_player* class), PDA device is an instance of *Digital_media* (i.e., which is instance from *Things* class), and *presevedby* is the inverse property of preserve property. *SN* is a functional property where its domain is the *PDA device* and its range is the *0G4023-32-362* (i.e., which is an instance of the *Literal* class). In addition, the forensic information resulted by the foresnic tool (e.g., AFF4) can also be represented in the *CF-CoC* framework using the same steps mentioned in Sections V.B and V.C.

D. Provenance of Information (Layer 4)

As we mentioned in Section II.C, the provenance metadata can be added to the terns during their design or to set of RDF triples. The framework CF-CoC use the named graph method to add provenance metadata to set of triples by naming them using URI.

Role players are responsible to add different provenance metadata to describe their forensic information. Each role player is responsible to provide complete and correct information about the origin and contents of his CoC(s) in order to be admissible in the court of law. In this context, data is shared on a small scale (i.e., LCD) between role players and juries. Identities of role players have been validated before the investigation process through exchanging of digital certificates. Thus, adding provenance metadata manually to the forensic information being published does not affect the creditability of these metadata. However, this is not the same case of sharing data on opened scale (i.e., LOD), where public data needed to be tracked and verified in order to ensure its creditability.

An example of metadata added to the level of terms presented in Figure 13, where the DC vocabulary is used to answer when the ontology is published and who published it. Provenance metadata can be attached during the phase of T-Box and A-Box.

Figure 20 is a good example to add provenance metadata using the named graph method. This figure represents the task of state preservation in the cyber forensic acquisition phase. Figure 6 provides abstract models for the named graph. The NG_{acqui} is the named graph of acquisition phase, which contains three tasks. One of them is the preservation task provided in Figure 20.

Figure 21 depicts how provenance metadata is added to a named graph. The CF-*CoC* assigns automatically the URL address to each ontology by adding a suffix NG to the ontology URL. For example, if the URL of acquisition ontology is https://127.0.0.1/Acquisition.rdf then it will be https://127.0.0.1/AcquisitionNG.rdf. In the same screen, the CF-*CoC* requires to select the ontology from which we will select the desired property from different provenance vocabulary (e.g., DC, FOAF, etc.).

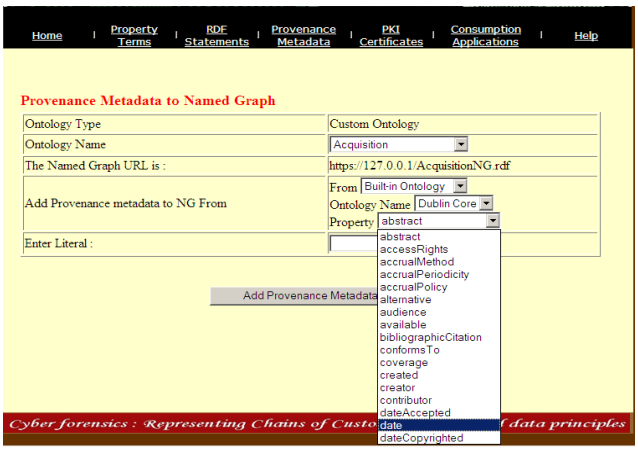

Figure 21. Add Provenance Metadata to Named Graph

As shown in Figure 21, the user selects the vocabulary/ontology in order to select the desired property, After selecting the property the user enters the literal representing the object. Thus, this is also considered as an RDF triple, where the subject is the URL address of the named graph of the acquisition phase, the predicate is the property of provenance vocabulary, and the object is the literal value.

E. PKI (Layer 6)

This section explains how to create the digital certificate using the four procedures mentioned above (see Figure 8).

Before creating the server and client certificate, a CA certificate will be created to sign both client and server requests (i.e., in this scenario, we will create manually a CA instead to buy it from a well-repudiated CA). Usually, the CA certificate is provided by a well-known CA provider (e.g., VeriSign Inc, Entrust Inc, etc.). In this scenario, a CA self-signed certificate is manually created.

The next part will present the set of theoretical procedure that juries and role players use together to share the digital certificates. This part will be followed by a detailed explanation of how these theoretical parts can be implemented and realized:

1. Juries send a list of players who are supposed to work on the current cyber crime case. Sending this list to the CA controls the data access to only these players. This prevents the disclosure (keeps the confidentiality) of data to unauthorized people.
2. The role player generates a public-private key pair ({KU-P, KR-P}), where P is all information identifying the player, R is private, and U is public. The player stores the private key in a secure storage to keep its integrity and confidentiality, and then sends the public key KU-P to the CA.
3. The player's public key and its identifying information P are signed by the authority using its ({KR-CA}) private key. The resulting data structure is back to the role player. R-CA {P, KU-P} is called the public key certificate of the role player, and the authority is called a public key certification authority (i.e., symbols outside brackets mean the signature of the data structure).
4. Juries obtain the authority's public key {KU-CA}.
5. Each player creating a *CoC* must authenticate himself to juries by signing his RDF graph G using his private key R-P{G} (i.e., all triples describing a phase are assembled in one graph called G). Later, before the court session, each player sends the certification R-CA {P, KU-P} to juries accompanied with the signed graph R-P{G}.

The next part will explain how such procedures can be implemented using the SSL tool:

1) Self-Signed Certificate:

Before starting, the CA key is generated, *RootCA.key* of length 2048 bits (2 bytes).

openssl genrsa –out RootCA.key 2048

The *RootCA.key* is then used to generate the certificate request *RootCA.csr* by providing the country name (i.e., C=CA), the organization name (i.e., O=Cyber Forensics Institution), and the common name of the certificate (i.e., CN=CF-CA) (see Figure 22).

openssl req -new -key RootCA.key -out RootCA.csr -config openssl.cnf -subj "/C=CA/O=Cyber Forensics Institution/CN=CF-CA/"

After generating the RootCA.csr, the request is signed using the RootCA.key to generate the requested certificate (*crt* format, *RootCA.crt*), but in this type of certificate, the CA itself will sign the certificate, that's why it is called self-signed certificate:

openssl req -x509 -days 365 –in RootCA.csr -out RootCA.crt -key RootCA.key -config opensslCA.cnf -extensions v3_ca

Finally, the exportable format p12 is generated to transform the *RootCA.crt* into an exportable format *RootCA.p12*

openssl pkcs12 -export -in RootCA.crt -inkey RootCA.key -certfile RootCA.crt -out RootCA.p12

Figure 22. CA self signed certificate

2) Server Certificate:

The server certificate is created for two goals: it lets the role player ensures the identity of the server, as well it is used to check for the client certificate (see Figure 23).

Assume that the server IP is corresponding to the server in [82]. This certificate will be issued for the juries to install it on their server. This server will host the *CF-CoC application*, which will be used by the role player. Thus, the CA will issue and sign a certificate for this IP name.

First, the *Server.key* is generated using the following command:

openssl genrsa -out Server.key 2048

The *Server.key* is then used to generate the certificate request *Server.csr* by providing the country name (i.e., C=CA), the organization name (i.e., O=Cyber Forensics Institution), and the common name of the certificate (i.e., CN=192.168.2.12).

openssl req -new -key Server.key -out Server.csr -config openssl.cnf -subj "/C=CA/O=Cyber Forensics Institution/CN=192.168.2.12/"

After generating the *Server.csr*, the request is signed using the CA certificate *RootCA.crt* and the key *RootCA.key* to generate the requested certificate (i.e., *Server.crt*).

openssl ca -days 365 -in server.csr -cert RootCA.crt –out Server.crt -keyfile RootCA.key -config opensslserver.cnf -extensions server

Because the server certificate is signed by the CA, the *openssl* command uses a build in parameter called '*ca*', to declare that the server certificate will be signed by the CA using its key (*RootCA.key*).

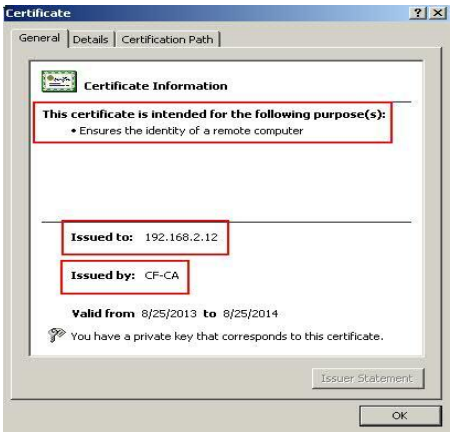

Figure 23. Server digital certificate

3) Client Certificate:

The role player authenticates himself to the server through the client certificate. Without this certificate, the role player will not be able to access *CF-CoC* application to construct different ontologies for each forensic phase and publish different resources (see Figure 24).

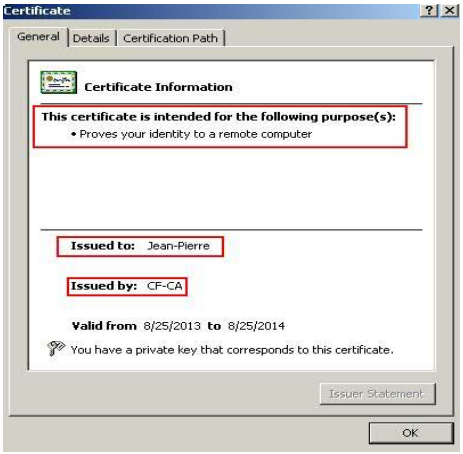

Figure 24. Client digital certificate

First, the *Client.key* is generated using the following commands:

openssl genrsa –out Client.key 2048

The *Client.key* is then used to generate the certificate request *Client.csr* by providing the country name (i.e., C=CA), the organization name (i.e., O=Cyber Forensics Institution), and the common name of the certificate (i.e., CN=Jean-Pierre).

openssl req -new -key Client.key -out Client.csr -config openssl.cnf -subj "/C=CA/O=Cyber Forensics Institution/CN=Jean-Pierre/"

After generating the *Client.csr*, the request is signed using the CA certificate (*RootCA.crt*) and key (*RootCA.key*) to generate the requested certificate (i.e., *Server.crt*).
openssl ca -days 365 -in Client.csr -cert RootCA.crt -out client.crt -keyfile RootCA.key -config opensslclient.cnf -extensions client

As shown in Figures 22, 23, and 24, we noticed that each certificate has its own purpose(s). Purpose(s) of a certificate depends on its type. The type of certificate is defined using the *-extension* in the creation of *crt* certificate. The *–extension* parameter calls the proper module for each certificate type. For example, it calls the *opensslCA.cnf*, *opensslServer.cnf*, and *opensslClient.cnf* for the CA, server, and client certificates, respectively. However, the *openssl.cnf* contains general configuration of all types of certificates.

4) Installation of Digital Certificates

Before installing the certificate, the CA sends to the jury and the role player their own certificates. Jury installs his certificate on his server and role player installs his certificate on his browser.

4.1) Self-Signed Certificate:

After creating the CA certificate, the CA sends to the server and client his certificate (i.e., p12 format without the private key of the CA certificate). By clicking on the p12 file (i.e., exportable format), a wizard will be launched to install the CA certificate in the trusted root folder of the current browsers for both server and client. By firstly installing this certificate on the server and client machines, their browsers will automatically identify the issuer of the client and server certificates.

4.2) Server Certificate:

The CA sends the server certificate to the jury. The latter then starts the installation of the server certificate. Installation of server certificates on Windows XP passes by two phases:

- Running the Microsoft Management Console and follow the steps in [99].
- Installing server certificates using the steps mentioned in [100].

4.3) Client Certificate:

Installing the client certificate is the same as the CA certificate, but at this time, the wizard installs the certificate in the client/ Personal folder of the browser.

5) Experimentation

This section shows how the scenario is enrolled after the role player and jury install their certificates:

- The client accesses the site by typing the URL of the server 192.168.2.12

- Because the remote server (i.e., where the *CF-CoC* web application is hosted) owns a server certificate, it requires then that his clients also owns a client certificate owned by the same trusted party (In this case, the CF-*CA*), otherwise the browser responded with a blank page (see Figure 25).

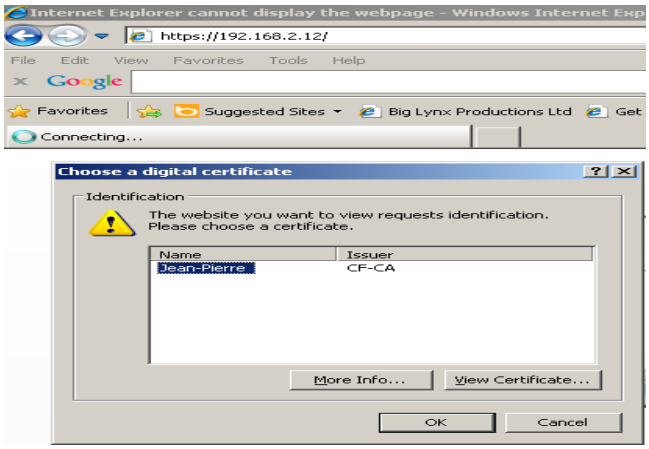

Figure 25. Server requires Client Digital certificate

- Once the server identifies the client certificate, it redirects the client to CF-*CoC* web application (see Figure 26).

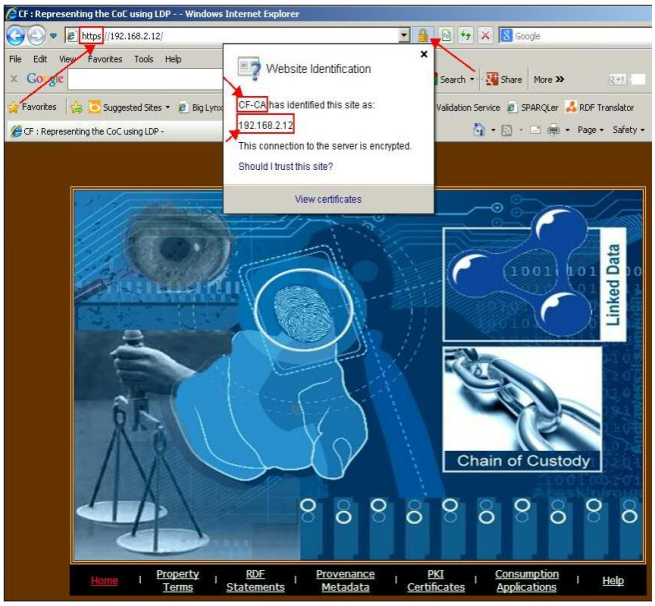

Figure 26. Redirection to the Restricted Resources

- Once the role player accesses the application, he starts to publish the ontologies and creates terms describing the forensic phase in hand (see the term definition module, Section V.B).

As we see in Figure 26, the server certificate is installed and shown in the top of the screen as a yellow lock. By clicking on the lock, it will show who issued the certificate (i.e., CA) for this page and to whom it was issued.

Once the role player finishes the publication task, the resources will be available to jury for consumption, as he owns a server certificate of the server, which allows him to view and access such resources published on his server. Resource as Jean-Pierre (see Figure 13) will be resolvable to more extra resources in the same domain [89] or to external domain. However, Jean-Pierre will not be accessible from external resources outer the former domain.

A certificate can not be created only for resources on the server but it can be issued for a specific resource on a server. For example, if we imagine that we have a resource 'x' in DS1, then the field of the certificate called 'issued to' (see Figure 23) will be assigned the complete URL of the resource 'x' (e.g., *CN=192.168.2.12/resources/x).*

F. Pattern Consumption Application (Layer 5)

After defining and publishing the foresnic information, juries can consume such resources uisng different pattern methods, whether by browsing the resources and navigate between different resoures, crawling using certain keyword, or by query the RDF triples.

The framework implemented some of them and imported the others. For example, the search and crawl was implemented, SPARQL endpoint is installed.

1) Browsing and crawling of resources

There exist many applications to browse and crawl RDF statements. All of them may have different consumption interfaces, but they are all common in the concept of how browsing and crawling are performed. In the CF-*CoC* framework, both types of consumption are simply presented by querying the RDF database and by standing on the defereneceable option to navigate between different RDF resources (see Figure 27).

Figure 27. Crawling and Browsing Consumption

Figure 27 shows the consumption screen of RDF statements. Juries can crawl a specific resources by selecting the first option and enter as keyword the required resources (e.g., Jean-Pierre), or through selecting the second option, search by forensic phase. In fact, the forensic phase

appearing to the juries is the same as the ontology terminology. As mentioned before (see Figure 4), each forensic phase is corresponding to an ontology, and each category corresponding to a forensic task in this forensic phase. For example, the preservation task is a category that contains different terms, the preserve verb (the task itself), what is the subject of this task, and who can perform this task, and what are the different ancestors of the term subject, predicate and objects.

If the user selects search by resource, he can enter a resource name to extract more information about such resource. The results of this type of search can lead to browse and dereference more related resources. For example, if the jury search for a resource called Jean-Pierre, its result will be: he was the creator of the acquisition phase. Jury in this case may get another deferenceable URL of the acquisition ontology, or another forensic task related to this forensic phase. This fact allows juries to navigate and discover more resources and retrieve more information related to all tasks performed by Jean-Pierre (see Figure 28).

Figure 28. Crawling using Jean-Pierre Resource

If the user selects the second option, he can select different ontologies that have been published by the role players, and he can select a specific task from selected ontology.

2) SPARQL Query

The SPARQL Language is the query language of the RDF. The SPARQL endpoint is installed on the local machine where the CF-*CoC* application resides.

Juries will not only able to query RDF triples, but also the provenance metadata associated to these triples. An example of how SPARQL queries the named graph and retrieve the provenance metadata such as publisher name and publishing date, is mentioned below:

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT  ?Publishername ?NamedGraph ?publicationdate
FROM NAMED <https://127.0.0.1/authenticationNG.rdf>
WHERE
{
        ?NamedGraph dc:publisher ?Publishername .
        GRAPH ?NamedGraph dc:date ?publicationdate }
}
```

Another example to query all RDF triples that are containing predicate foaf:name, and cf-coc:preserve is shown below:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX cf-coc: <http://www.127.0.0.1/vocab/acquisition#>
 SELECT * from <http://127.0.0.1/Acquisition>
 WHERE
 {
      ?person foaf:name ?person_name .
      ?person  cf-coc:preserve ?media_name .
 }
```

As shown from above examples, query of RDF triples necessitate from juries the awareness of semantic parts and technical skills to write SPARQL code. In this case, the consumption of RDF data using SPARQL query language is not appropriate for juries to be one of their consumption patterns. Juries are specialized in law and legal procedures, not in the field of information technology. Thus, the need of a module that can reason over the RDF triples is required to be implemented. This will avoid that the juries need to be aware about this technical knowledge and proficient the SPARQL query code. This module will be based on different semantic rules of RDFS and the primitives of OWL.

SPARQL query language can not only query explicitly the RDF triples, but it is also able to infer triples that are not physically stored. This advantage resides on SPARQL when the latter has a rule base that can be used to infer implicit and hidden information.

In our context, LD is lightweight ontology using RDFS and some primitives of OWL. RDFS has some inference rules and reasoning for its constructors (see Table IV).

For example, the screen below shows a reasoning on owl:FunctionalProperty and owl:InverseFunctionalProperty of proprietary terms ‘preserve’ and ‘SN’, respectively.

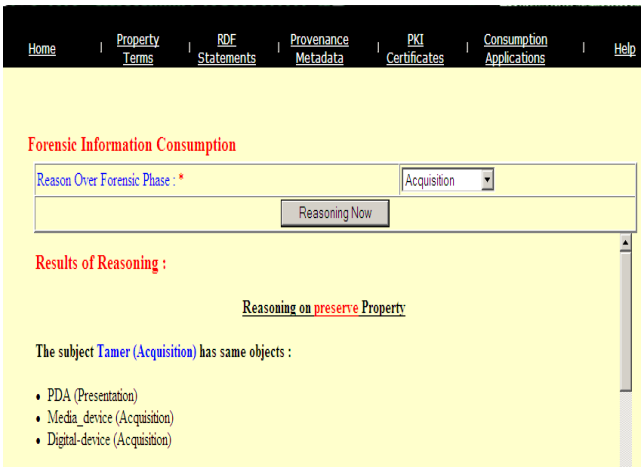

Figure 29. Reasoning on preserve Property

As shown in Figure 29, the juries select the ontology that he wants to reason. In this figure, they selected the acquisition phase (i.e., ontology), which contains the

property term called ‘preserve’ (i.e., the property of this term is tagged to be FunctionalProperty).

By referring to Table VI, we notice that if a property p is tagged to be FunctionalProperty, then all objects containing the predicate ‘preserve’ and having same subjects (i.e., Tamer), are equivalent to each other (i.e., PDA, Media_device, and Digital_device are equivalents).

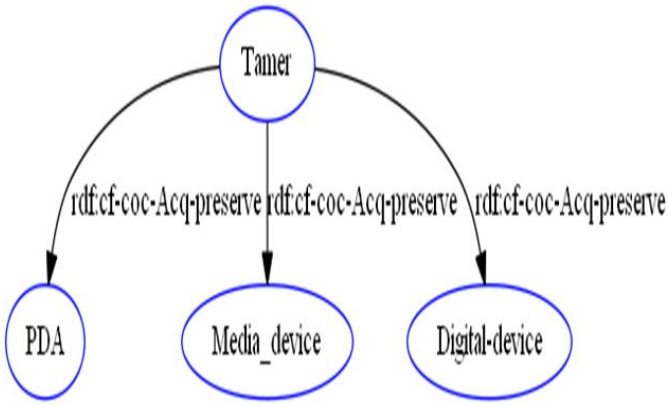

Figure 30. RDF Triples of preserve Property

Resource URL : https://127.0.0.1/Vocab/Acquisition#preserve

Term Type :	Property , Functional Property
Sub Property of :	made
Domain :	First_Responder
Range :	Digital_media
Inverse of :	preservedby

Below are the instances of class First_Responder	Predicate	Below are the instances of class Digital_media
Tamer (Acquisition Phase)	cf-coc-Acq-preserve	PDA (Presentation Phase)
Hamdi (Investigation Phase)	cf-coc-Acq-preserve	Computer (Acquisition Phase)
Tamer (Acquisition Phase)	cf-coc-Acq-preserve	Media_device (Acquisition Phase)
Hamdi (Investigation Phase)	cf-coc-Acq-preserve	Laptop (Acquisition Phase)
Tamer (Acquisition Phase)	cf-coc-Acq-preserve	Digital-device (Acquisition Phase)

Figure 31. Deferenceability of preserve Property

Again, by referring to Table VI, we notice that if a property ‘p’ is tagged to be InverseFunctionalProperty, then all subjects containing the predicate SN and having same objects (i.e., T1-236-185F), are equivalent to each other (i.e., Iphone and PDA, see Figures 32 and 33).

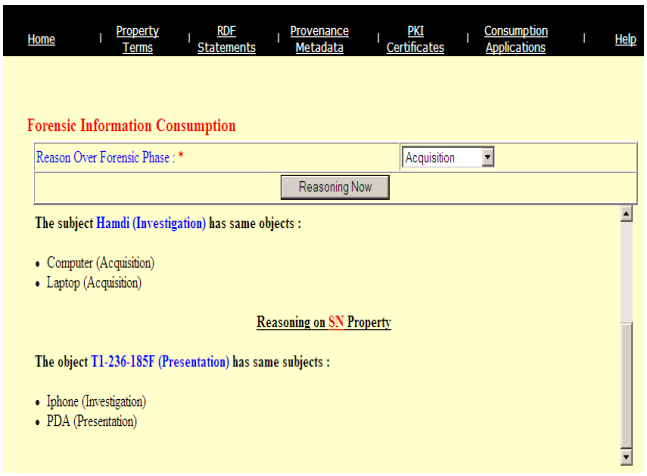

Figure 32. Reasoning on SN Property

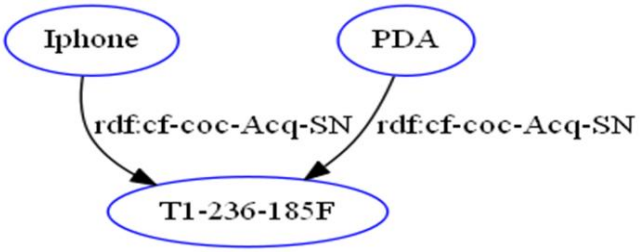

Figure 33. RDF Triples of SN Property

VI. CONCLUSION

This paper depicted a novel framework that will be used by role players to represent tangible *CoC* resulted from their cyber investigation. Role players use this framework to represent and publish forensic resources in order to be consumed by juries in the court of law.

This work explained in details all layers of the framework that are based on the technology stack of the LD. This technology stack (RDF, HTTP, and URI) is used to represent and publish different resources on the web in a structured way.

The role players start their representation process by defining new proprietary/custom terms describing the forensic information of their tangible *CoC*. This task is performed using lightweight ontology through the RDFS constructors and some primitive from OWL. Role players may also accompany different provenance metadata imported from the vocabularies of the semantic web to describe the origin of forensic information and strength the trustworthiness with the juries of court. All represented resources are then published in RDF format upon URI resolution, in order to be shared in a small scale between the role players and juries through the public key infrastructure approach. The latter opens the door to a new era of research representing the counter part of the linked open data, called

the linked closed data, which share all the advantages of the LOD, but with consumption restriction.

This work elaborated and explained the framework through the preservation task imported from the acquisition phase of the Kruse model. In future work, the framework will obey empirical experimentations through different scenarios imported from different forensic phases. This will be accomplished by defining all tasks for each phase of a forensic process. Different scenarios can be provided from different forensic models. In addition, supplementary layers may be added to this framework to facilitate communication between users (i.e., role players and users) and the framework. For example, the addition of an intelligent layer that transforms data between end-user and data store, and intelligent tutor layer that can guide the role players to use different well defined semantic vocabularies helps to define proprietary terms and their constraints, and learns role players the way of publishing data using such vocabularies. (i.e., in case they do not have enough technical knowledge about the LD).

REFERENCES

[1] T. F. Gayed, H. Lounis, and M. Bari, "Linked closed data using PKI: a case study on publishing and consuming data in a forensic process," The Sixth International Conference on Advanced Cognitive Technologies and Applications (IARIA 2014), Venice, Italy, pp. 77-86, ISBN: 978-1-61208-340-7.

[2] E. Kenneally, "Gatekeeping out of the box: open source software as a mechanism to assess reliability for digital evidence," Virginia Journal of Law and Technology, vol. 6, no. 13, issue 3, Fall 2001.

[3] T F. Gayed, H. Lounis, and M. Bari, "Representing and (im)proving the chain of custody using the semantic web," The Fourth International Conference on Advanced Cognitive Technologies and Applications (IARIA 2012), Nice, France, pp. 19-23, ISBN: 978-1-61208-218-9.

[4] M. W. Andrew, "Defining a process model for forensic analysis of digital devices and storage media," Proceedings of the 2nd International Workshop on Systematic Approaches to Digital Forensic Engineering (SADFE 2007), pp. 16-30, ISBN: 0-7695-2802-2, Seattle, WA.

[5] M. Köhn, J. H. P. Eloff, and Ms. Olivier, "UML modeling of digital forensic process models (DFPMs)," Proceedings of the International Security South Africa (ISSA 2008), Innovative Minds Conference, Johannesburg, South Africa, pp. 1-13, Jul. 2008.

[6] Electronic Crime Scene Investigation: A Guide for First Responders, Second Edition, 78 Pages, by Independent Publishing Platform; 2 Edition (July 9, 2012), ISBN-10: 147827683, ISBN-13: 878-1478276845.

[7] E. Casey, "Digital Evidence and Computer Crime - Forensic Science Computers and the Internet," 3rd Edition, Academic Press 2011, pp. 1-807, ISBN: 978-0-12-374268-1.

[8] S. O. Ciardhuain, "An extended model of cyber crime investigations," International Journal of Digital Evidence, vol. 3, issue 1, 2004.

[9] Y. Yusoff, R. Ismail, and Z. Hassan, "Common phases of computer forensics investigation models," International Journal of computer science and information technology (IJCSIT 2011), vol. 3, no. 3, pp. 17- 31.

[10] T. F. Gayed, H. Lounis, and M. Bari, "Computer forensics: toward the construction of electronic chain of custody on the semantic web," Proceedings of the 24th International Conference on Software Engineering & Knowledge Engineering (SEKE 2012), pp. 406-411.

[11] T. F. Gayed, H. Lounis, and M. Bari, "Cyber forensics: representing and managing tangible chain of custody using the linked data principles," The international conference on Advanced Cognitive technologies and Application (IARIA 2013), pp. 87-96, ISSN: 2308-4197, ISBN: 978-1-61208-273-8, Valencia, Spain.

[12] T. F. Gayed, H. Lounis, and M. Bari, "Representing chains of custody along a forensic process: a case study on Kruse model," Proceedings of the 25th International Conference on Software Engineering & Knowledge Engineering (SEKE'2013), pp. 674-680, ISBN 1-891706-33-0 ISSN: 2325-9000.

[13] G. C. Kessler, "Judges' Awareness, Understanding and Application of Digital Evidence," PhD Thesis in Computer Technology in Education, Graduate school of computer and information sciences, Nova Southeastern University, 2010.

[14] RDF Model and Syntax Specification, W3C recommendation, 22 Feb 1999, www.w3.org/TR/REC-rdf-syntax-19990222/1999 [retrieved Oct. 2014].

[15] D. Brickley and R. V. Guha. "RDF Schema," W3C Recommendation http://www.w3.org/TR/rdf-schema/ [retrieved Nov. 2014].

[16] OWL: web ontology language overview, W3C Recommendation, http://www.w3.org/TR/owl-features/ 2004 [retrieved Nov. 2014].

[17] O. Hartig, "Provenance information in the web of data," In proceedings of the linked data on the web (LDOW 2009), Workshop at the World Wide Web Conference (WWW), Madrid, Spain.

[18] Dublin Core Metadata Initiative: http://dublincore.org/documents/dcmi-terms/ [retrieved Nov. 2014].

[19] FOAF Vocabulary Specification 0.99: http://xmlns.com/foaf/spec/ [retrieved Nov. 2014].

[20] P. P. da Silva, D. L. McGuinness, and R. Fikes, "A proof markup language for semantic web services," Information Systems, pp. 381–395, vol. 31, issue 4, Jun. 2006.

[21] T. Berners-Lee, R. Fielding, and L. Masinter, "RFC 2396 – Uniform Resource identifiers (URI): Generic Syntax," http://www.isi.edu/in-notes/rfc2396, Aug 1998 [retrieved Feb. 2013].

[22] R. Fielding, "Hypertext transfer protocol," – http/1.1 request for comments:http://www.w3.or/Protocols/rfc2616/rfc2616.html, 1999 [retrieved Mar. 2013].

[23] T. Omitola et al., "Tracing the provenance of linked data using voiD," The International Conference on Web Intelligence, Mining and Semantics (WIMS 2011), vol. 17, no. 17, ISBN: 978-1-4503-0148-0.

[24] L. T. Berners-Lee, "Design issues: linked data," from http://www.w3.org/DesignIssues/LinkedData.html [retrieved Feb. 2013].

[25] M. Cobden, J. Black, N. Gibbins, L. Carr, and N. R. Shadbolt, "A research agenda for linked closed dataset," In Proceeding of the 2nd International Workshop on Consuming Linked Data (COLD 2011), vol. 782 [vision paper].

[26] Linking Open Data, W3C Semantic Web Education and Outreach (SWEO) Community Project, http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData [retrieved: Oct. 2014].

[27] C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data— The story so far," International Journal on Semantic Web and Information Systems (IJ-SWIS 2009), vol. 5, no. 3, pp. 1-22.

[28] T. F. Gayed, H. Lounis, and M. Bari, "Creating proprietary terms using lightweight ontology: a case study on acquisition phase in a cyber forensic process," Proceedings of the 26th International Conference on Software Engineering & Knowledge Engineering (SEKE 2014), Vancouver, Canada, pp. 76-81.

[29] T. Berners-Lee, J. Hendler, and Ora Lassilia, "The semantic web," Scientific American, vol. 5, May 2001, pp. 34–44.

[30] RDF/XML Specifications: W3C Recommendation 10 February 2004, http://www.w3c.org/TR/REC-rdf-syntax [retrieved Oct. 2014].

[31] Turtle – Terse RDF Triple Language: W3C Team Submission 14 January 2008, http://www.w3.org/TeamSubmission/turtle/ [retrieved Nov. 2014].

[32] Resource Description Framework Anchor (RDFa) in Extensible Hyper Text Markup Language (XHTML): Syntax and Processing,

W3C Recommendation 14 Oct. 2008, http://www.w3.org/TR/rdfa-syntax/ [retrieved Nov 2014].

[33] RDF 1.1 Test Cases: http://www.w3.org/TR/2014/NOTE-rdf11-testcases-20140225/ [retrieved Nov. 2014].

[34] Notation3 (N3): A Readable RDF Syntax: W3C Team Submission 14 January 2008, http://www.w3.org/TeamSubmission/n3/ [retrieved Sep. 2014]

[35] L. M. Campbell and S. MacNeill, "The semantic web, linked and open data, a briefing paper," Centre for Educational Technology, Interoperability and Standards (JISC CETIS 2010), SN: 2010:B03.

[36] K. Alexander, R. Cyganiak, M. Hausenblas, and J. Zhao, "Describing Linked Datasets - On the Design and Usage of VoiD, 'the Vocabulary Of Interlinked Datasets'," WWW 2009 Workshop: Linked Data on the Web (LDOW 2009), Madrid, Spain.

[37] P. Hitzler and F. V. Harmelen, "A reasonable semantic web," vol. 1(1), 2010, pp. 39-44, URL: http://corescholar.libraries.wright.edu/cse/25 [retrieved Sep. 2014].

[38] G. Antoniou and F. V. Harmelen, "Web ontology language: OWL," Handbook on Ontologies 2004, pp. 67-92.

[39] S. Vanstone, Oorschot, and A. Menezes, "Handbook of Applied Cryptography," CRC Press, 1997, ISBN: 0-8493-8523-7.

[40] C. LT. Brown, "Computer Evidence: Collection & Preservation," 1ST Edition, ISBN: 0-619-13120-9, 2006.

[41] T. Health and C. Bizer, "Linked Data: Evolving the Web into a Global Data Space," http://linkeddatabook.com/editions/1.0/ [retrieved: Nov. 2014].

[42] D. Berrueta and J. Phipps, "Best Practice Recipes for Publishing RDF vocabularies," - w3c note. http://www.w3.org/TR/swbp-vocab-pub/, 2008 [retrieved Nov. 2014].

[43] N. Mendelsohn, "The Self-Describing Web," http://www.w3.org/2001/tag/doc/selfDescribingDocuments.html, 2009 [retrieved Sep. 2014].

[44] A. Brinson, A. Robinson, and M. Rogers, "A Cyber Forensics Ontology: Creating a New Approach to Studying Cyber Forensics," The International Journal of Digital Forensics & Incident Response (DFRWS 2006). vol. 3, pp. 37-43.

[45] Simple Knowledge Organization System: http://www.w3.org/2004/02/skos/ [retrieved Oct. 2014].

[46] W. Kruse II and J. Heiser, "Computer Forensics: Incident Response Essentials," Addison Wesley, 2002, ISBN-13: 978-0201707199, ISBN-10: 0201707195, 1st Edition.

[47] OWL Web Ontology Language Guide: http://www.w3.org/TR/owl-guide/ [retrieved: Nov. 2014]

[48] G. Palmer, "A road map for digital forensic research," Technical Report from the First Digital Forensic Research Workshop (DFRWS), Utica, New York, 2001, DTR – T001-01 Final.

[49] B. D. Carrier, "A Hypothesis-based approach to digital forensic investigations," PhD thesis, Center for Education and Research in Information Assurance and Security, Purdue University, West Lafayette, IN 47907-2086.

[50] B. Carrier, "Defining digital forensic examination and analysis tool using abstraction layers," International Journal of Digital Evidence IJDE 2003, vol. 1, issue 4, pp. 1-12.

[51] E. Politcnica, "Improving chain of custody in forensic investigation of electronic digital systems," International Journal of Computer and Networks Security (IJCNS 2011), vol. 11, no. 1, pp. 1-9.

[52] J. Cosic and M. Baca, "A framework to (im)prove chain of custody in digital investigation process," Proceedings of the 21st Central European Conference on Information and Intelligent Systems (CECIIS 2010), pp. 435-438, Varaždin, Croatia.

[53] J. Cosic and M. Baca, "(Im)proving chain of custody and digital evidence integrity with timestamp," 33rd Proceedings of the International Convention on Information and Communication Technology Electronics and Microelectronics, ICT Convention (MIPRO 2010), Opatija, pp. 1226-1230, ISBN: 978-1-4244-7763-0.

[54] R. Jueneman and R. Lapedis, "Solving the digital chain of custody problem," Trusted Mobility Solutions, SPYRUS 2010, Document number 412-000001-02 [white paper].

[55] A. Bogen and D. Dampier, "Knowledge discovery and experience modeling in computer forensics media analysis," In International Symposium on Information and Communication Technologies (ISICT 2004), pp. 140-145, ISBN: 1-59593-170-8.

[56] B. Schatz, PhD thesis, "Digital Evidence: Representation and Assurance," Faculty of Information technology, Queensland University of Technology, Oct. 2007.

[57] B. Schatz, G. Mohay, and A. Clark, "Rich event representation for computer forensics," Proceedings of the 17TH International Conference in Knowledge Based and Intelligent Information and Engineering Systems (KES 2013), vol. 22, pp. 1266-1275, ISBN: 0-9596291-9-1.

[58] B. Schatz, G. Mohay, and A. Clark, "Generalising event forensics across multiple domains," Proceedings of the 2004 Australian Computer Network and Information Forensics Conference (ACNIFC 2004), pp. 136-144, Perth, Australia.

[59] S. Al-Fedaghi and B. Al-Babtain, "Modeling the forensic process," International Journal of Security and its Application, Vol. 6, no. 4, Oct. 2012, pp. 79-107.

[60] Common Digital Evidence Format (CDESF), Available from: http://www.dfrws.org/CDESF/index.html [retrieved Nov. 2014].

[61] S. L. Garfinkel, D. J. Malan, K-A. Dubec, C.C. Stevens, and C. Pham, "Disk imaging with the advanced forensics format, library and tools," Advances in Digital Forensics, 2nd Annual IFIP WG 11.9, In Proceeding of International Conference on Digital Forensics 2006, Orlando, Florida.

[62] P. Turner, "Unification of Digital Evidence from Disparate Sources (Digital Evidence Bags)," In 5th Digital Forensic Research Workshop (DFRW 2004), vol. 2, issue 3, pp. 223-225, New Orleans.

[63] M. Cohen, M. Garfinkel, and B. Schatz, "Extending the advanced forensic format (AFF) to accommodate multiple data sources, logical evidence, arbitrary information and forensic workflow," Digital Investigation: The International Journal of Digital Forensics and Incident (IJDFI 2009), vol. 6, pp. 57-68, ISSN: 1742-2876.

[64] Simple Knowledge Organization System RDF Schema (SKOS) Vocabulary: http://www.w3.org/TR/2008/WD-skos-reference-20080829/skos.html [retrieved Nov. 2014].

[65] Open Provenance Model Vocabulary Specification: http://purl.org/net/opmv/ns [retrieved Oct. 2014].

[66] O. Hartig and J. Zhao, "Publishing and consuming provenance metadata on the web of linked data," International Provenance and Annotation Workshop (IPAW 2010), LNCS 6378, vol. 6378, Berlin, pp. 78-90, ISBN 978-3-642-17819-1.

[67] How to Publish Linked Data on the Web: http://www4.wiwiss.fu-berlin.de/bizer/pub/linkeddatatutorial/ [retrieved Oct. 2014].

[68] L. Moreau et al., "The open provenance model core specification," (v1.1), Future Generation Computer. Systems., vol. 27, issue 6, 2011, pp. 743-756.

[69] J. J. Carroll, C. Bizer, P. Hayes, and P. Stickler, "Named graphs, provenance and trust," In Proceeding of WWW '05 of the 14th international conference on World Wide Web (WWW), USA, ACM Press, pp. 613-622, ISBN: 1-59593-046-9.

[70] R. Isele, A. Harth, J. Umbrich, and C. Bizer, "Ldspider: An open-source crawling framework for the web of linked data," In 9th International Semantic Web Conference (ISWC 2010) & Demonstrations Trash: Collected Abstracts vol. 658.

[71] Describing Linked Datasets with the VoID: http://www.w3.org/TR/void/ [retrieved Mar. 2014].

[72] E. Rajabi, M. Kahani, and M. Angel Silicia, "Trustworthiness of Linked Data Using PKI," World Wide Web Conference (www2012) Lyon, France, 2012.

[73] J. Davies, "Implementing SSL/TLS Using Cryptography and PKI," Indianapolis, Indiana: Wiley Publishing Inc, ISBN: 978 -0-470-92041-1, 2011.

[74] D. Richard, V. C. Hu, W. Timothy, and S. Chang, "Introduction to public key technology and the federal PKI infrastructure," Technical Report, SP 800-32, National Institute of Standards and Technology (NIST), U.S. Government publication, 2013 Edition.

[75] M. Blaze, J. Feigenbaum, and A. Keromytis, "KeyNote: Trust management in the public key infrastructure," 6th International Workshop Cambridge, UK, vol. 1550, pp. 59-63, ISBN: 978-3-540-49135-4 January 1999 [White paper].

[76] E. Barker et al., "Recommendation for Key Management Part 3: Application-Specific Key Management Guidance," NIST Special Publication 800-57, 2013 Edition.

[77] Public Key Infrastructure, Entrust: www.entrust.com/what-is-pki/ [retrieved: Feb. 2014].

[78] R. Perlman, "An overview of PKI trust models, In IEEE network," vol. 13, issue 6, pp. 38-43,1999, ISSN: 0890-8044.

[79] Extended Validation SSL Certificate: The Next Generation High Assurance SSL Certificate, http://www.evsslcertificate.com/ssl/description-ssl.html [retrieved: Sep. 2014].

[80] Internet X.509 Public Key Infrastructure Certificate Management Protocols: https://tools.ietf.org/html/rfc2510 [retrieved: Mar. 2014].

[81] Official Site of OpenSSL Project, http://www.openssl.org/ [retrieved: Dec. 2013].

[82] Cyber Forensics-Chain of Custody Server Host, Domain owned by Tamer Gayed, www.cyberforensics-coc.com [retrieved: Oct. 2013].

[83] D. Quan and D. R. Karger, "How to make a semantic web browser," WWW'04 Proceedings of the 13th International Conference on World Wide Web, pp. 255-265, New York, USA, ISBN: 1-58113-844-X.

[84] Semantic Web Search Engine (SWSE): http://www.swse.org/ [retrieved Oct. 2014].

[85] Semantic Web Search (Swoogle): http://swoogle.umbc.edu/ [retrieved Sep. 2014].

[86] L. Davis, H. Shrobe, and P. Szolovits, "What is a knowledge representation?," AI Magazine, 1993, vol. 14(1), pp. 17-32.

[87] J. Zhao, C. Bizer, Y. Gil, P. Missier, and S. Sahoo, "Provenance requirements for the next version of RDF," A position paper based on the work of the W3C Provenance Incubator Group, Standford, CA, June 2010.

[88] B. Glimm, A. Hogan, M. Krötzsch, and A. Polleres, "OWL: yet to arrive on the web of data?," In Proceeding of Linked Data on the Web Workshop (LDOW 2012), vol. 937, Lyon, France.

[89] A. Polleres, A. Hogan, R. Delbru, and J. Umbrich "RDFS & OWL Reasoning for Linked Data," Semantic Technologies for Intelligent Data Access, Lecture Notes in Computer Science, vol. 8067, pp. 91-149. ISBN: 978-3-642-39783-7.

[90] Easy RDF: http://www.easyrdf.org/ [retrieved: Nov. 2014].

[91] Graphiz – Graph Visualization Software: http://www.graphviz.org/Documentation.php [retrieved: Feb. 2013].

[92] Internet Information Services, Microsoft, http://www.iis.net/ [retrieved: Mar. 2014].

[93] Official Site of OpenSSL Project, http://www.openssl.org/ [retrieved: Dec. 2013].

[94] T. Heath, "How will we interact with the web of data?," IEEE Internet Computing 2008, vol. 12, issue 5, pp. 88–91, ISSN: 1089-7801.

[95] T. Berners-Lee et al., "Tabulator: Exploring and analyzing linked data on the semantic web," In Proceedings of the 3rd International Semantic Web User Interaction Workshop (SWUI 2006), Athens, Georgia.

[96] O. Hartig, C. Bizer, and J. Christoph Freytag, "Executing SPARQL queries over the web of linked data," In Proceedings of the 8th International Semantic Web Conference (ISWC 2009), vol. 5823, pp. 293–309, ISBN: 978-3-642-04930-9.

[97] G. Cheng and Y. Qu, "Searching linked objects with falcons: Approach, implementation and evaluation," International Journal on Semantic Web and Information Systems (IJSWIS 2009), pp. 49–70.

[98] http://inference-web.org/2007/primer/ [retrieved: Jan. 2013].

[99] Server Certificate Installation Instructions, Microsoft Developer Network: http://msdn.microsoft.com/en-us/library/ms751408.aspx [retrieved: Feb. 2014].

[100] IIS Management Microsoft Management Console (MMC), Microsoft, http://support.microsoft.com/kb/892987 [retrieved: Feb. 2014].