Secure Semantic Web Application Development: Present and Future

Umara Noor[1], Zahid Rashid[2]
National University of Science and Technology
Pakistan
13phdunoor@seecs.edu.pk
zahid190@gmail.com

ABSTRACT: *Semantic web is the future realization of the current web, in which resources are annotated with machine understandable meta-data, allowing the automation of tasks by employing these resources in their correct contexts. Several notable technologies have been specified by the World Wide Web Consortium (W3C) for the application development lifecycle of semantic web. The worldwide adoption of these technologies requires the development of secure semantic web applications. Therefore the research community has proposed to define and integrate security standards to all phases of semantic web application development. For this purpose the research community has reported some of the preliminary efforts in the form of XML Key Management (XKMS), Security Assertion Markup Language (SAML), XML Access Control Markup Language (XACML) and Platform for Privacy Preferences (P3P). But there is a need to develop a security assessment benchmark for semantic web applications. The effective assessment of security of semantic web applications has been paid less attention so far in this regard. For this reason in this research work we propose some important features related to the development of a benchmark for secure semantic web application development that can effectively assess security of semantic web applications based on the survey of the security aspect of important semantic web languages i.e. XML, RDF, OWL and SPARQL/ SPARUL. The aim of the benchmark is to facilitate developers in building secure semantic web applications and provide security professionals to develop sophisticated security tools by effective performance evaluation.*

Keywords: Security, Semantic web, Benchmark, XML, RDF, OWL, SPARQL/ SPARUL

Received: 19 May 2016, Revised 27 June 2016, Accepted 3 July 2016

1. Introduction

The advent of WWW has changed the traditional ways of communication and businesses. Computing devices have turned into gateways for huge information reservoirs. To access these information reservoirs web search tools serve an important role. Today's web search tools suffer with serious limitations of high recall and low precision, low or no recall, result sensitivity to vocabulary and lack of information integration from multiple sources. Despite several improvements in search/ information retrieval technology, the problem is still there. The major cause of which is due to the fact that a large part of existing web content is significant to humans for its consumption and is solely controlled by its own application. The problem was first revealed by the father of WWW, Tim Berners Lee with the solution of making web content machine accessible. Such a kind of web was termed as semantic web.

Semantic web is a web of machine understandable data, provided with a common framework that allows data to be shared and reused across application, enterprise, and community boundaries [1]. The aim of the semantic web is to automate the tasks of planning, coordination, search and integration by developing knowledge management systems and software agents capable of:

- Organizing knowledge in conceptual spaces according to its meaning
- Knowledge maintenance by checking for inconsistencies
- Query based knowledge extraction/ integration
- Implementing security standards according to the defined policies
- Performing electronic commerce (B2C, B2B)

The realization of semantic web has led to evolution in many areas of web technologies. These technologies are mainly concerned with effective web content management and retrieval. Figure 1 shows the layer cake for semantic web as described in [2]. The tasks of each layer are explicitly specified. The World Wide Web Consortium (W3C) is providing standards for each layer in the form of specifications.

At the bottom we have Unicode and URI for encoding data and identifying resources on the web. TCP/ IP, HTTP and SSL are used as data transmission protocols on this layer. This layer does not deal with the syntax and semantics of the web document. The next layer deals with XML, XML schemas and namespaces. XML provides a way to represent data in a structural way so that the web documents can have a uniform representation and presentation when exchanged over web. XML schemas define the structure of the XML documents.

Now XML deals only with syntax of the document. In order to deal with the semantics RDF was proposed. RDF provides a way for representing information about resources in the World Wide Web [3].The information is represented in the form of graph which is built using statements. A statement is an expression comprised of subject-predicate-object relation. The subject and object are web resources and predicate represent property of the subject. Such statements are also termed as triples. Now RDF statements can have different interpretations. The appropriate interpretation is defined by RDF schema which can be thought of as a vocabulary for RDF.

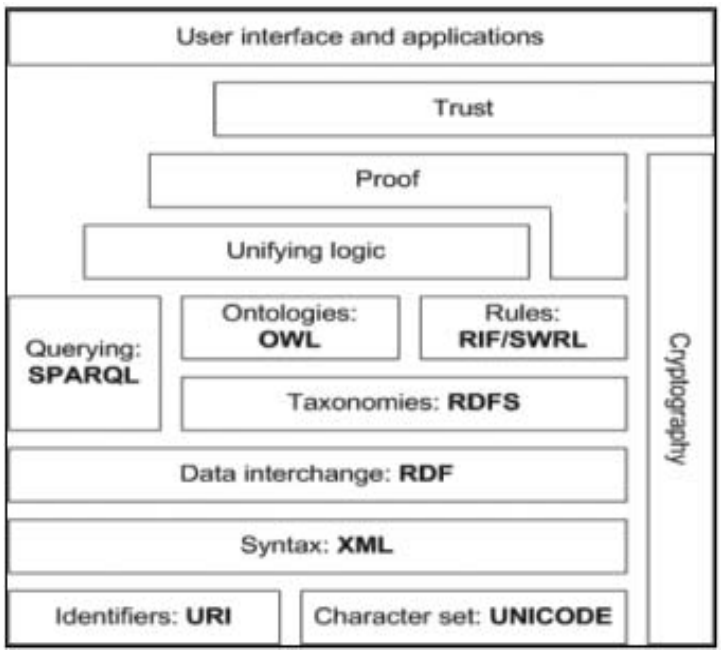

Figure 1. Semantic Web Layer Cake

Next to RDF is the ontology layer. Ontology is an explicit and formal specification of a conceptualization [4]. Generally ontology describes a domain of discourse in a formal way. Ontology comprises of a finite number of terms and relationships between them. The terms indicate important concepts of the domain. The relationships establish hierarchies among these concepts. Simple taxonomies can be defined using RDF schema but for modeling complex ontologies other more powerful languages have been developed. Such languages include DAML+OIL, OIL and OWL. The ontology definition is accompanied with the rules and querying capabilities. SPARQL has been proposed as a standard query language for querying RDF graphs.

The last three layers are logic, proof and trust. Currently these layers are not fully defined. However we can have an anticipation of the tasks performed by them. The aim of the logic layer is to provide an explanation for the tasks performed by lower layers. The proof layer actually defines the logic deduction process. Finally the task of trust layer can be viewed from two dimensions. From one dimension it defines the trustworthiness of a source and from the other dimension it is related to defining security policies for secure access of semantic web sources.

2. Background

Semantic web technologies are constantly evolving and are widely deployed in different kinds of web environments including web applications, web services and e-businesses. Such environments have highly flexible and user friendly interfaces, which makes them significantly vulnerable to attacks by malicious users. Further the adoption of some powerful features of semantic web like integration and interoperability without security can also lead to unwanted aftereffects. Therefore security for semantic web technology has become an important consideration in recent years in order to gain its true benefits. Here in this section we provide an overview of the security issues for semantic web. Several security standards have been devised for the secure operations of semantic web. A detailed discussion of the developing and envisioned security standards is followed in the literature review.

Regarding making semantic web secure, security should not be considered as a separate entity. This means that security is not the job of any single layer of the semantic web paradigm but it should be integrated to all the layers [5]. Now we describe layer-wise security challenges pose to semantic web. On the lowest layer end-to-end security is required which is achieved through several legacy methods defined for network security. On the XML and XML schema layer we need to control access to various segments of web documents for read and update. On the RDF and RDF schema layer we need to determine the true interpretation and context of the document for various scenarios. Then the next step is to make ontology layer secure. Ontologies may have security levels defined for its different parts. The process of integration and interoperability performed through querying should be secure. Here security can also be introduced at the rule level. As the logic, proof and trust layers are related to inference capabilities of the entire system therefore the problem of unauthorized inferences should be prevented. Along with the security another important consideration is that of flexibility and ease of use. As the intent of semantic web technologies as envisioned by its developers was to give power to the web through intelligence. This power should not be compromised with the implementation of security measures. So security should be defined at different levels and degree for different applications.

3. Security In Semantic Web Languages

Along with developing specifications for each layer of semantic web and getting all of its benefits for semantic information retrieval it has also become important to analyze its security aspect. Currently security assessment for semantic web is in a phase which is devoid of many clues on which human society have believed till now for security assessment. Here we discuss about those security standards of semantic web paradigm that are proposed so far or are in some stage of development.

3.1 XML Security

XML is a standard representation language for document exchange that lets one write structured web documents with a user-defined vocabulary. With its growing worldwide demand for document exchange, it has become important to define multiple security solutions for it. Basically security standards for XML have been addressed from two aspects. The first one defines XML security as a set of security techniques and the second one deal with models and languages specifying and exchanging access control policies [8]. The key challenge of both aspects is to provide integrity, confidentiality and access control to entire XML document or some part of it in a way that does not comprise XML's intrinsic flexibility property.

The core XML security standards as defined by W3C include digital signature for providing integrity and signatures; encryption for providing confidentiality; XML Key Management (XKMS) for providing public key registration, location and validation;

Security Assertion Markup Language (SAML) for conveying authentication, authorization and attribute assertions; XML Access Control Markup Language (XACML) for defining access control rules and policies; and Platform for Privacy Preferences (P3P) for defining privacy policies and preferences [9].

3.2 RDF Security

RDF is the specification developed by W3C for specifying the semantics of entities, events and relationships found in web documents in order to have a common interpretation across multiple source integration. Currently no standards exist for defining the security aspect of RDF. However a few preliminary approaches provide us with a roadmap towards RDF security [5], [10]. In these approaches security for RDF has been discussed with respect to four issues. One is to use RDF for specifying security policies; the second is to ensure access control for RDF documents; the third is to investigate the use of RDF for the secure interoperability of resources and services on the web, and the last one deals with developing techniques for the secure publications and dissemination of RDF documents.

In the paper [10], authors discuss major issues involved in RDF security and then exploit semantic richness of RDF for policy specification on a general scenario and from the general scenario the implied authorizations are automatically entailed. One of the main issues involved in RDF security is that how can access control policies be specified on a given RDF description of a domain. So there is a need to introduce extensions to RDF for policy specification. Secondly the interoperability between databases and services is also essential. For this purpose the security policy specified in RDF also needs to be integrated so that an integrated policy can be designed for this web of linked data. In past considerable work has been done on security in XML documents, their secure publishing and dissemination. Hence there is a need to explore the applicability of techniques used for XML security in RDF security. Since semantics of documents are involved here, RDF documents will require additional mechanisms.

The protection of a web resource starts with the definition of rules that govern access to it. For example, consider a rule that states that only the creator of a resource is able to modify it. Each policy is implemented in the form of tuples of the type <s,o,m> which defines the rule that a subject s can have access to object o, under the condition that it possesses access mode Secure Interoperability/ Integration.

The next to RDF layer is the ontology layer, where ontology languages enable users to write explicit, formal conceptualizations of domain models. One major feature of semantic web is semantic data integration from diverse and disparate data sources. All the data may not be in databases. It can exist in the form of structured or unstructured files, in the form of tables, text, images, audio and video. Semantic web aims to integrate data not only semantically but also securely encountering the property of quality for data in the form of trusted sources and trusted request.

Some of the ontology languages are: DAML+OIL and OWL [11] OWL is the specification developed by W3C. OWL has more facilities for expressing meaning and semantics than XML, RDF, and RDF-S, and thus OWL goes beyond these languages in its ability to represent machine interpretable content on the Web. OWL is a revision of the DAML+OIL web ontology language. Sublanguages of OWL are OWL full, OWL DL, and OWL Lite.

Currently no security standard exists for OWL. But the research community have proposed / outlined the directions towards achieving it. Security on this layer can be viewed from two perspectives: defining ontologies for secure semantic information integration and defining ontologies to describe security threats e.g. defining ontologies for firewalls and intrusion detection system [12] and defining semantic policy language and use of distributed policy management as an alternative to traditional authentication and access control schemes [13].

3.3 SPARQL/ SPARUL Security

With RDF the information on the web is represented in the form of directed, labeled graph of data. SPARQL is the language specially designed to query RDF data. SPARQL is basically based on SQL and is a read only language. A companion version of SPARQL is SPARQL/ update that allow describing, communicating and storing updates to an RDF store. Security on this layer of semantic web is viewed by the research community from the perspective of threats and attacks.

Query languages like SQL, LDAP and XPath have been susceptible to attacks based on non-sanitized user inputs. The vulnerability becomes obvious when such inputs are directly concatenated to query strings. This helps attackers to directly control the execution of query which results in an unwanted behavior in the application. Such attacks can be prevented by providing

libraries with the appropriate tools to sanitize user inputs in these languages. Extensive work has been reported to prevent code injection in these languages but currently no significant activity has been reported for semantic web query languages except [14]. In which the possibility of code injection attacks for SPARQL/ SPARUL have been analyzed with a few examples. Initially three kinds of attacks are observed:

1) SPARQL Injection

It is an attack in which malicious code is inserted into strings that are later passed to an instance of server for parsing and execution.

2) Blind SPARQL Injection

Blind SPARQL injection is identical to normal SPARQL injection except that when an attacker attempts to exploit an application, rather than getting a useful error message, they get a generic page specified by the developer instead. An attacker can still steal data by asking a series of true and false questions through SPARQL statements.

3) SPARUL Injection

These attacks can change the meaning of the query. SPARUL introduces the chance to modify the ontology by the use of INSERT, DELETE and MODIFY statements.

4. Existing Semantic Web Benchmarks

Benchmarks are one important aspect of performance evaluation. Currently several benchmarks have been proposed for semantic web in the domain of reasoning, interoperability storage mechanisms and language assertion [15-20]. Here we provide a brief overview of the existing benchmarks.

LUBM [15] is an application specific benchmark for semantic web knowledge base systems. The benchmark enables one to choose the appropriate KBS for large OWL application by evaluating the reasoning capabilities and storage mechanisms of the KBS. Thus provides a means of measuring trade-off between scalability and reasoning. Now there are two basic requirements of large OWL applications: 1) scalability and efficiency and 2) sufficient reasoning capabilities. To fulfill these requirements LUBM provides ontology for the university domain, synthetic OWL data scalable to an arbitrary size, a set of fourteen extensional queries representing a variety of properties and several performance metrics. The performance metrics are: load time, repository size, query response time, query completeness and soundness and a combined metric to measure the tradeoff. Two memory-based and two persistent storage systems have been evaluated with LUBM. LUBM benchmark serves as a guideline for developers in advancing state of the art in semantic KBSs and in developing benchmarks for other domains of semantic web technology.

SP2Bench [16] is a language-specific benchmark proposed for SPARQL performance evaluation. The goal of SP2Bench is to test most common SPARQL constructs, operator constellations and a broad range of RDF data access patterns which is generally not possible through application specific benchmarks. The language-specific benchmark requires data to be in some representative format and challenging queries should be designed in order to effectively evaluate the performance. Therefore the benchmark is evaluated against DBLP dataset. DBLP data is encoded in RDF data format and several challenging queries can be generated for it. SP2Bench comprises of:

A data generator to support the creation of meaningful arbitrarily large DBLP documents in RDF format, reflecting key characteristics and social-world relations found in the original DBLP database. The generated documents cover various RDF constructs, such as blank nodes and containers.

A set of 17 benchmark queries carefully designed to test a variety of operator constellations, data access patterns, and optimization strategies.

The performance metrics for SP2Bench are success rate, loading time, pre-query performance, global performance and memory consumption.

Berlin SPARQL [17] provides a benchmark to compare the performance of storage systems that expose SPARQL end points via the SPARQL protocol. The goal of the benchmark is to assist application developers in choosing the right architecture and the

right storage system for their requirements. The performance of two kinds of storage systems is compared i.e. native RDF store and SPARQL to SQL rewriters. The evaluation is performed against four popular RDF stores (Sesame, Virtuoso, Jena TDB, and Jena SDB) with the performance of two SPARQL-to-SQL rewriters (D2R Server and Virtuoso RDF Views) as well as the performance of two relational database management systems (MySQL and Virtuoso RDBMS).The benchmark also provide means of revealing strengths and weaknesses of current RDF stores and SPARQL-to-SQL rewriters which might be useful for the developers in order to improve them in future.

To judge the performance of a reasoning system and techniques natural workloads are not always readily available. Data generators of existing systems may lead to meaningless workloads and thus effective performance of the reasoners cannot be judged. Therefore in [18], a benchmark for generating meaningful ontology workload for semantic web reasoning engines is developed. In which synthetic ontologies are generated which serve as workload for standard reasoning benchmarks. To construct synthetic ontologies, DAML ontology library is statistically analyzed and important parameters are derived.

5. Web Application Security

Due to the versatility of HTTP, web browsers have become entry point for nearly all kinds of interactions with the internet. Similarly the exciting features of web 2.0 have made web applications an indispensible platform for delivering services, e-commerce, trade and social networking activities. All these best features of web 2.0 can be compromised and lead to disastrous results in the absence of appropriate security measures. Thus web application security has emerged as an important sub-domain of information security in the recent years and has been paid much attention by the industry along with other kinds of security measures such as network security and operating system security.

It has been observed that web application developers spend most of the time on features, functionality and SEO techniques of the web application and thus security is totally ignored or paid less attention as required. The major reasons behind this attitude are the lack awareness on the part of developer and time constraints on the part of the project management hierarchy.

Web applications provide an entry point for the most critical resources of the corporate i.e. web servers and database servers. A vulnerable hole in the web application may cause a malicious intruder to gain access to web server and database server. After gaining access, the intruder can spoil a web site, insert spam links which direct users to another site, insert malicious code that installs itself onto the user's computer, insert malicious code that steals session IDs (cookies), steal user's information and browsing habits, steal account information, steal information stored in the database and access restricted content.

Most common web application attacks as observed by [6] and [21] are: SQL injection, XSS (Cross Site Scripting), remote command execution and path traversal. Here we provide a brief description for two very popular of web application attacks.

5.1 SQL Injection

SQL injection is a code injection technique that attacks a web database by taking advantage of the vulnerabilities present in the interface of a web application. This is done by passing malicious SQL commands as input to web form entries. The outcomes of the attack include changes or deletion of highly sensitive business information, stealing of user's personal information such as social security number, credit card numbers and spoiling brand credibility. SQL injection attack can also be used with other kinds of attacks to cause more damage to a web application.

SQL injection attacks are currently on the top of two highly popular web application threat ranking systems i.e. OWASP [6] and CWE/ SANS [21].

5.2 Cross Site Scripting

Cross site scripting can be thought of as a special case of code injection attack. It is done by inserting a malicious client-side script into a web application by bypassing security mechanisms of web browsers. Such scripts are usually inserted in the input areas like forum and comments section readily available on social networking web sites. When the user visits that web site a single click on the script starts executing it. The execution of these scripts cause installation of malicious software on the victim's computer, account hijacking through stealing victim's cookies, or hijacking of the victim's session. Another damage caused by cross site scripting includes web site defacement and vandalism.

6. Proposed Features of Secure SEmantic Web Application

As we saw in the previous sections that semantic web technologies are developing at a constant space and are being adopted widely. One of the important aspects that need to be considered is the development of secure semantic web applications. As we observed in the case of non-semantic web applications that insecurity can compromise the important corporate assets. Similarly in case of semantic web applications, all the functional powers of semantic web are compromised along with corporate assets. In view of this, all security measures should be encountered from the first step of semantic web application lifecycle. This requires the developers to be trained regarding the security issues of semantic web.

Now it is not possible for each developer to develop full blown semantic web applications and then scan it for vulnerabilities. Also, it is not a viable solution for those security professionals who frequently need to test security tools against a vulnerable platform to validate its performance as advertised. For this reason here in our research work we discuss some important features of a security evaluation benchmark for semantic web applications.

As currently the research area for semantic web security is in its infancy that's why we don't have a big list for the kind of attacks a semantic web application is vulnerable to. The features of the benchmark are therefore classified among two hierarchies: the first part concentrates around the existing security standards and the second part concentrate around the identification of new threats and attacks. As the most dangerous attack considered so far for non-semantic web application is the SQL injection attack. For this reason in case of semantic web applications we assume SPARQL injection attacks to be the most dangerous attacks. Following is a list of preliminary features defined for the benchmark:

1. The proposed benchmark should enable users to learn and practice existing security standards defined to ensure integrity, confidentiality, access control policies and privacy for the XML, XML schema layer.
2. The user is able to learn and practice policy implementation for secure interoperability and integration of ontologies.
3. The users should be allowed to identify new threats and attacks in safe and legal environment.

An environment for learning and practicing code injection attacks (XML injection, SPARQL, Blind SPARQL and SPARUL injection) will be provided to the users.

7. Conclusion And Future Work

There is an ongoing rapid evolution in World Wide Web from human understandable forum to machine understandable Web, named as Semantic Web. The ability of machines to read and understand the Web content is made possible by the Resource Description Framework, which is based on resources, attributes and relationships. The current World Wide Web is prone to a large number of threats, viruses and worms. The advent of Semantic Web brings with itself security challenges that are far greater than today's Web, with increased chances of security breaches. So it is critical to consider security issues of Semantic Web, as we develop it. In this research work, we surveyed current semantic web security approaches for developing a secure semantic web application present on XML, RDF, OWL and SPARQL/ SPARUL related layers. Then we discussed the importance of some benchmarking tool to assess semantic web application's security aspect. We also proposed some features for the benchmark. In our future work we will develop a benchmark to assess semantic web application security based on the features proposed in this paper.

References

[1] Web Primer, The Semantic Web. http://www.w3schools.com/web/web_semantic.asp

[2] Antoniou, G., Harmelen, F. V. (2004). A Semantic Web Primer, MIT Press.

[3] Frank Manola, Eric Miller, Brian McBride (editors): RDF Primer, W3C Recommendation 10 February 2004. http://www.w3.org/TR/2004/REC-rdf-primer-20040210/

[4] Gruber, Thomas R.(1993). A translation approach to portable ontology specifications, *Knowledge Acquisition*, 5 (2) 199–220.

[5] Thuraisingham, B. M.(2005). Security standards for the semantic web. *Computer Standards & Interfaces* 27 (3) 257-268

[6] Michael Coates (OWASP Chair): Welcome to OWASP, the free and open software security community. https://www.owasp.org/index.php/Main_Page

[7] Michael Coates (OWASP Chair): OWASP Web goat project. https://www.owasp.org/index.php/OWASP_WebGoat_Project

[8] Ardagna, C., Damiani, E., De Capitani di Vimercati, S., Samarati, P., (2007) XML Security, *In*: Security, Privacy and Trust in Modern Data Management 6. 71-86.

[9] Frederick Hirsch (chair), XML Security Working Group. www.w3.org/2008/xmlsec/

[10] Carminati, B., Ferrari, E., Thuraisingham, B. M., Using RDF for Policy Specification and Enforcement, DEXA Workshops 2004: 163-167

[11] Deborah L., McGuinness, Frank van Harmelen (editors): OWL Web ontology language overview, W3C Recommendation 10 February 2004. http://www.w3.org/TR/owl-features/

[12] Vorobiev, A., Han, J.(2003). Security Attack Ontology for Web Services. SKG 2006: 42 L. Kagal, T. W. Finin, A. Joshi: A Policy Based Approach to Security for the Semantic Web. International Semantic Web Conference 2003: 402-418

[13] Orduna, P., Almeida, A., Aguilera, U., Laiseca, X., Lopez-de-Ipina, D., and Goiri, A.G., Identifying Security Issues in the Semantic Web: Injection Attacks in the Semantic Query Languages, JSWEB, Spain, 2010, http://www.morelab.deusto.es/index.php/publications-1879995610/2010

[14] Guo, Y., Pan, Z., and Heflin, J., (2005) LUBM: A Benchmark for OWL Knowledge Base Systems, *Journal of Web Semantics*, vol (3) 158–182.

[15] Schmidt, M., Hornung, T., Lausen, G., Pinkel, C., (2009) SP2Bench: A SPARQL Performance Benchmark, ICDE, 222–233, IEEE.

[16] Bizer C., Schultz, A., (2009) The Berlin SPARQL Benchmark, *International Journal of Semantic Web Information Systems* 5 (2) 1–24.

[17] Tempich, C., Volz, R., Towards a benchmark for Semantic Web reasoners - an analysis of the DAML ontology library. EON 2003

[18] Morsey, M., Lehmann, J., Auer, S., and Ngomo, A.N., (2011) DBpedia SPARQL Benchmark –Performance Assessment with Real Queries on Real Data, The Semantic web-ISWC, Springer.

[19] Garcia-Castro, R., Gómez-Pérez, A., Large-Scale Benchmarking of the OWL Interoperability of Semantic Web Technologies. ICSC 2008: 214-221

[20] Martin, Bob., Brown, Mason., Paller, Alan., Kirby, Dennis., Christey, Steve (editors) (2011). Common Weakness Enumeration, 2011 CWE/SANS Top 25 Most Dangerous Software Errors. http://cwe.mitre.org/top25/