Tag-based Information Flow Analysis for Document Classification in Provenance

Jyothsna Rachapalli Murat Kantarcioglu Bhavani Thuraisingham

The University of Texas at Dallas

Abstract

A crucial aspect of certain applications such as the ones pertaining to Intelligence domain or Health-care, is to manage and protect sensitive information effectively and efficiently. In this paper, we propose a tagging mechanism to track the flow of sensitive or valuable information in a provenance graph and automate the process of document classification. When provenance is initially recorded, the documents of a provenance graph are assumed to be annotated with tags representing their sensitivity or priority. We then propagate the tags appropriately on the newly generated documents using additional inference rules defined in this paper. This approach enables users to conveniently query to identify sensitive or valuable information, which can now be efficiently managed or protected once identified.

1 Introduction

Information leaks are a serious problem and in particular confidential data leaks are considered as one of the worst kinds of leaks which could cause serious damage to organizations. Therefore, it becomes imperative to detect and manage such leaks efficiently. If the flow of sensitive information is tracked using suitable tags, one can easily detect if the leaked information was sensitive by checking its tag. Depending on the sensitivity of the leaked information identified by the tag, a user can expedite damage assessment and take relevant damage control measures. For example, in the case of Intelligence domain, a user could use the tag based provenance information to track the flow of sensitive information and efficiently secure and manage sensitive information. The Tagging mechanism can not only help in identifying the sensitivity of leaked data but also help with sanitization and access control decisions. For example in the health care domain, tagging can enable an application to detect whether the given document contains sensitive information and decide whether it can be shared and with whom can it be shared.

Provenance data may often grow enormous in size giving rise to a need for it to be scoped. Provenance needs to be scoped according to the user's interest; otherwise, by default, the provenance of any item would conceptually trace back to the Big Bang, marking the origin of the Universe [5]. One of the things a scope can identify is the type of source data the user is interested in. In a given application domain a user can define a set of tags representing the relative importance of the data they are attached to. If the source data is tagged suitably when the provenance is recorded initially, the tags can be propagated, using the inference rules, indicating the flow of the data in the system from the various sources. This can help with many application specific decisions such as recording fine-grained provenance information for data labeled with high priority tags and recording minimum required provenance information for data labeled with low priority tags. The following summarizes our contributions:

- We define inference rules of the Tagging mechanism based on the Open Provenance Model (OPM) [6]. OPM is a general model of provenance that is designed to allow provenance information to be exchanged between systems (facilitating interoperability), by means of a compatibility layer based on a shared provenance model and defines provenance in a precise, technology-agnostic manner. We build our tagging mechanism based on OPM, in order to make it more general and independent of any specific domain or technology such as databases, workflows or distributed systems.
- We then propose an implementation of the Tagging mechanism using Web Ontology Language (OWL) and Semantic Web Rule Language(SWRL).

2 Tag based Information flow analysis

In the first subsection we propose the inference rules for tag propagation in an OPM graph. Subsequently, we describe an implementation of the rules using OWL and SWRL in the second subsection.

2.1 Tag Propagation Rules for OPM graph

For interoperability purposes, OPM defines a set of common properties which are used to construct an annotation instance that can be attached to any annotable entity. We define a new property called *tag* in addition to the aforementioned common properties. The subject of property *tag* is an artifact and its value is an integer as shown in the table below. The integer value on an artifact indicates its rating or relative importance when compared to other artifacts.

subject:	an artifact
property:	http://openprovenance.org/property#tag
value:	an Integer
meaning:	Represents Artifact rating or priority

OPM works on the assumption of absence of internal knowledge of the processes. Therefore, in order to develop a tagging mechanism based on OPM, one needs to work with the dataflow oriented view of provenance. The dataflow oriented view comprises of artifacts and "was derived from" edges connecting them. In the following, we state two rules based on the "was derived from" edges to propagate tags. There are essentially two cases. In the first case, an artifact is known to be derived from only one other artifact and in the second case, the artifact is known to be derived from more than one artifact. The following are the two rules for tag propagation:

- Single Source Derivation: If Artifact A1 is annotated with tag "t" and artifact A2 was derived from A1 then artifact A2 is annotated with tag t as well.
- Multi-Source derivation : If artifact A was derived from artifacts A1, A2, A3, ..., An, which are respectively annotated with tags "t1", "t2", "t3", ..., "tn", then artifact A is annotated with tag tx, where tag tx is the tag with the highest priority among the list of tags on the source artifact nodes.

In this paper we define tag propagation rules to track the flow of sensitive information. However, based on the application requirements the rules can be suitably modified. That is, a domain expert can restate how the tag is propagated in the case of Single source and Multi-source derivations. In fact a one can define tags of any other data type, such as string data type, in which case one is only required to ensure total order among string tag values.

2.2 Implementation with OWL and SWRL

We use Jun Zhao's OPMV [4] (Open Provenance Model Vocabulary) OWL ontology, which is a lightweight vocabulary to describe the core concepts of OPM. We create an OWL Datatype property called *tag*, whose domain is class *artifact* (class opmv:Artifact) and range is an integer value (xsd:integer). An instance of class opmv:Artifact is annotated using the property tag.

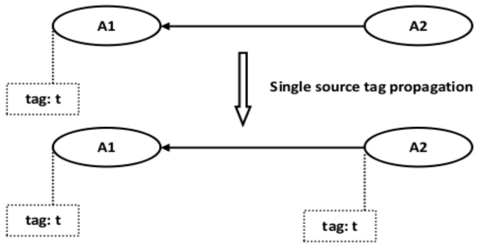

Figure 1: Tag propagation for Single Source Derivation

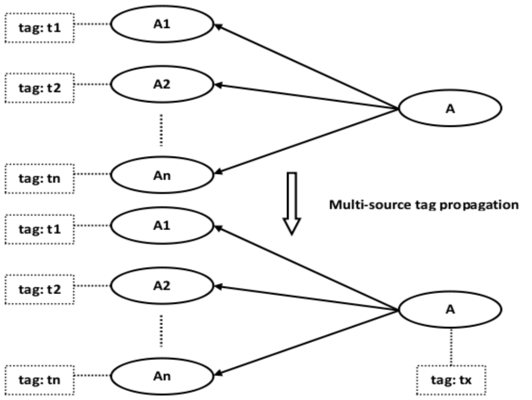

Figure 2: Tag propagation for Multi-Source Derivation

tag:	an OWL property
Identifier:	http://purl.org/net/opmv/ns#tag
OWL Type:	DatatypeProperty
Domain:	opmv:Artifact
Range:	xsd:integer

We now describe the SWRL rules for propagation of the tags. To propagate the tag in the case of single source derivation, we can use the following SWRL rule (given in human-readable syntax). It states that if there are two artifacts represented by variables ?a1 and ?a2 such that the artifact ?a1 has tag ?t and if ?a2 was derived from ?a1 then the artifact ?a2 is assigned tag ?t as well.

Artifact(?a1) ˆ Artifact(?a2) ˆ tag(?a1, ?t)
ˆ wasDerivedfrom(?a2, ?a1) → tag(?a2, ?t)

However, to create a rule for the case of multi-source derivation is not as straight forward since the number of source artifacts n, that are used for the derivation of the new artifact is not fixed. Since n can take any value, a naive brute force approach would be to set an upper limit for n and write a rule for all possible cases. That is, there would be n-1 rules corresponding to the cases for: derivation from two sources, three sources and so on up to derivation from n sources. However, a simpler

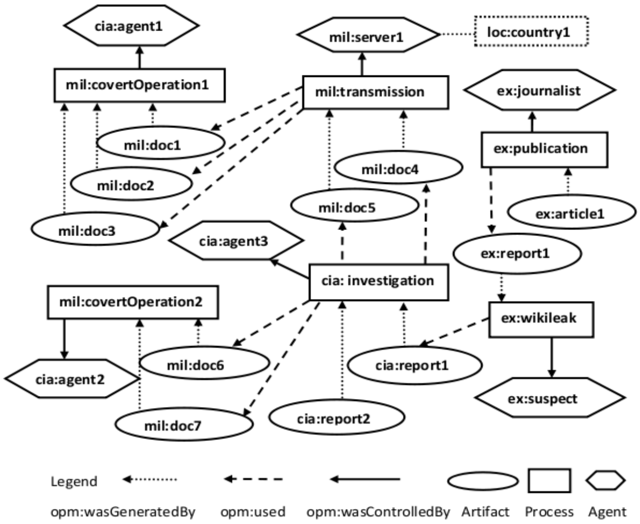

Figure 3: Intelligence Domain Workflow

alternative would be to use the rule described below in conjunction with the single source derivation rule.

Artifact(?a1) ^ Artifact(?a2) ^ wasDerivedfrom(?a2, ?a1)
^ tag(?a1, ?t1) ^ tag(?a2, ?t2)^ swrlb:greaterThan(?t1, ?t2)
→ tag(?a2, ?t1)

The multi-source derivation rule states that if ?a1 and ?a2 are artifacts such that ?a2 is derived from ?a1 and the tag on ?a2 (?t2) is smaller than the tag on ?a1 (?t1) then, tag value on ?a2 is updated to ?t1. Figure 2 illustrates multi-source derivation of artifact A from artifacts A1, A2, ..., An. When triple "A wasDerivedFrom A1" is encountered, the single source derivation rule annotates artifact A with a tag value the same as that of A1's tag value. When the next triple is encountered, say "A wasDerivedFrom A4", and if the current tag value of A is found to be smaller than the tag value of A4 (?t4), then it is replaced/updated with ?t4. Eventually, when all the n source derivations are considered, the artifact A is annotated with a tag whose value is highest among the source Artifact tag values.

3 Illustration of Tagging mechanism with Intelligence Domain Usecase

Although the tagging mechanism described is applicable to various domains, we illustrate the tagging mechanism with a usecase from the Intelligence domain, inspired by the usecase from [2], as it is more comprehensible. Figure 3 shows an example workflow from the Intelligence domain. It shows that process mil:covertOperation1 was controlled by cia:agent1 and the process generated mil:doc1, mil:doc2 and mil:doc3. The mil:transmission process, controlled by mil:server1 located in country1, used those three documents and in-turn generated documents mil:doc4 and mil:doc5. The process mil:CovertOperation2 was controlled by cia:agent2 and it generated documents mil:doc6 and mil:doc7. The process cia:investigation controlled by cia:agent3, used documents numbered 4, 5, 6 and 7 and produced documents cia:report1 and cia:report2. The workflow shows publication of topsecret information contained in document ex:article1 by agent ex:journalist. In this scenario, it becomes essential to quickly identify the nature of data-loss, perform damage assessment and take damage control measures.

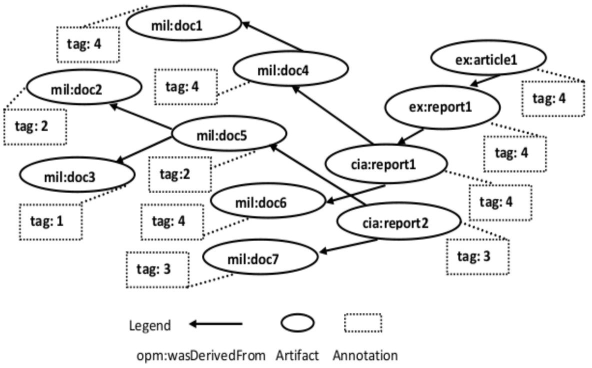

Figure 4: Dataflow Oriented View of the Workflow

The documents belong to four different levels of classification namely: Top secret, Secret, Confidential and Unclassified. A Top secret document is annotated with tag with value 4, a Secret document with value 3, a Confidential document with value 2 and an Unclassified document with value 1. As the workflow proceeds, processes in the workflow take as input, documents of various levels of classification and perform computations to generate new documents, which now need to be appropriately classified. This will further facilitate many important tasks such as enabling access control decisions. An interesting fact to observe is that the notion of data is not the same as the notion of artifact as was pointed out by [6]. We attempt to track the flow of sensitive data through the various documents.

Figure 4 shows the dataflow oriented view of the workflow given in figure 3. Initially, documents mil:doc1, mil:doc2 and mil:doc3 are annotated with tags 4, 2 and 1 respectively, indicating that the classification of mil:doc1 is Top Secret, mil:doc2 is Confidential and mil:doc3 is Unclassified. Since, mil:doc4 was derived from mil:doc1, using the single source derivation rule, mil:doc4 is annotated with tag with value 4. Similarly, using the multi-source derivation rule, mil:doc5 is annotated with tag value 2 as it is the maximum among its source tag values. Furthermore, cia:report1 is tagged with value 4 as it was derived from two top secret documents and cia:report2 is tagged with value 3 as it was derived from a secret document and a confidential document. In the following, we illustrate how SPARQL [7] queries (stated abstractly here) can leverage the tagging mechanism to quickly and efficiently compute the results:

- List all the topsecret documents
  SELECT ?x WHERE {?x opm:tag “4” }
- List all the topsecret documents generated within time interval t1 and t2.
  SELECT ?x
  WHERE {?x opm:tag “4” . ?x opm:time ?t.
  FILTER{ (?t >= t1) && (?t <= t2) }}
- What is the classification level of mil:doc1
  SELECT ?y WHERE{mil:doc1 opm:tag ?y}
- List the all the documents from which cia:report1 was derived along with their classification levels
  SELECT ?y ?z
  WHERE{cia:report1 opm:wasDerivedFrom ?y .
  ?y opm:tag ?z.}

Since a document tagged as topsecret is crucial, one can further annotate it with very detailed metadata or history of derivation and usage. A SPARQL query can then be used to retrive the metadata of the topsecret document and thus aid in tasks such as quickly making damage assesments and identifying potential suspects.

4 Related Work

In [3], the authors define SPARQL query templates to answer common queries such as why-provenance, where-provenance and how-provenance. The Tagging mechanism can be used to make these queries more expressive and enable answering of a broader range of queries. Implicit provenance approach in [1] considers the semantics of a query language, nested relational calculus, where a value has been annotated, with a color, denoting the origin of that value. As values are propagated, the language passes along annotations. When a result is produced by a program, the associated annotations indicate where the value was derived from. A similar approach is also adopted by [8], which introduces a formalism for provenance in distributed systems based on the $\pi-$calculus. It essentially annotates all data products with metadata representing their provenance. Here, annotations consist of sequences of send and receive events, that are extended whenever values are communicated by the application. [8]’s annotations are richer than [1]’s coloring scheme, which means that more sophisticated provenance queries can be answered. However, both the approaches address domain specific problems, whereas the tagging mechanism proposed by us is more general as it is based on OPM, which aims to capture provenance of information flowing across multiple systems facilitating interoperability.

5 Conclusion and Future Work

In this paper we have described a tagging mechanism to automatically classify documents generated in a provenance graph. It enables selective provenance by identifying and tracking the data that is most crucial or sensitive so that more resources can be invested to protect them. A fine-grained provenance can be recorded, consisting of the metadata such as the entire derivation history of an artifact, for the most crucial of the documents facilitating quick reference. In future, we envision a framework for tagging providing a sophisticated means for recording provenance. It will comprise of comprehensive mechanisms to tag the data as required by the application domains. The framework will also comprise of a list of queries supported by a given tagging mechanism based on its purpose. We would further like to explore how tagging can be leveraged for making decisions in access control workflows. We plan to extend the tagging mechanism, by allowing a given document to be annotated with multiple tags, which will separate the sensitive information from the non-sensitive within a document.[1]

Notes

[1] This work was partially supported by Air Force Office of Scientific Research MURI Grant FA9550-08-1-0265, National Institutes of Health Grant 1R01LM009989, National Science Foundation (NSF) Grant Career-CNS-0845803, and NSF Grants CNS-0964350, CNS-1016343, CNS-1111529.

References

[1] BUNEMAN, P., CHENEY, J., AND VANSUMMEREN, S. On the expressiveness of implicit provenance in query and update languages. In *ICDT* (2007).

[2] CADENHEAD, T., KANTARCIOGLU, M., AND THURAISINGHAM, B. A frameworkwork for policies over provenance. In *TaPP* (June 2011), USENIX.

[3] CADENHEAD, T., KHADILKAR, V., KANTARCIOGLU, M., AND THURAISINGHAM, B. M. A language for provenance access control. In *CODASPY* (2011), ACM.

[4] HARTIG, O., AND ZHAO, J. Provenance vocabulary core ontology specification, 2010.

[5] MOREAU, L. The foundations for provenance on the web 1, 2009.

[6] MOREAU, L., CLIFFORD, B., FREIRE, J., FUTRELLE, J., GIL, Y., GROTH, P., KWASNIKOWSKA, N., MILES, S., MISSIER, P., MYERS, J., PLALE, B., SIMMHAN, Y., STEPHAN, E., AND DEN BUSSCHE, J. V. The open provenance model core specification (v1.1). *Future Generation Computer Systems (FGCS) 27* (2011).

[7] PRUD’HOMMEAUX, E., AND SEABORNE, A. SPARQL Query Language for RDF. W3C Recommendation, 2008.

[8] SOUILAH, I., FRANCALANZA, A., AND SASSONE, V. A formal model of provenance in distributed systems. In *TaPP* (2009), USENIX.