



Signing individual fragments of an RDF graph

G.Tummarello, C.Morbidoni, P.Puliti, F.Piazza

Summary

Being able to determine the provenience of statements is a fundamental step in any SW trust modeling.

Previous proposals have been mostly "graph centric" where signatures cover entire graphs. [1] [2]

In this work, we designed and propose a methodology where digital signatures are written in the RDF itself and cover subsets of the entire graph.

Groups of statements signed with this methodology can be exchanged and safely inserted into any existing RDF triple store without the loss of provenience information.

This methodology has been implemented and is both available as open source library and deployed in a SW P2P project called DBin (www.DBin.org). Based on this methodology DBin implements trust filtering and information revocation (non monotonic behaviours) based on monotonic P2P RDF exchanges. [5]

What is the minimum piece of RDF graph that can be exchanged in a SW P2P scenario?

MSG = Minimum Self-contained Graph

Definition 1. An RDF statement *involves* a name if it has that name as subject or object.

Definition 2. An RDF graph *involves* a name, if any of its statements involves that name.

Definition 3. Given an RDF statement *s*, the Minimum Self Contained Graph (MSG) containing that statement, written MSG(*s*), is the set of RDF statements comprised of the following:

1. The statement in question;
2. Recursively, for all the blank nodes involved by the statements included in the description so far, the MSG of all the statements *involving* such blank nodes;

Theorem 1. If *s* and *t* are distinct statements and *t* belongs to MSG(*s*), then MSG(*t*) = MSG(*s*).

Theorem 2. Each statement belongs to one and only one MSG.

Corollary 1. An RDF model has an unique decomposition in MSGs.
→ A graph can be transferred incrementally one MSG at a time

Corollary 2. An MSG is identified by any of its statements
→ The MSG Signature can be attached to a single arbitrary triple

Signing Procedure

From the MSG definition, it comes that it is possible to attach "MSG scoped metadata" to any of its statements, e.g. via reification. An MSG signature is one such metadata. From a user Private Key, the MSG signature is created by hashing a string obtained by canonical RDF serialization as illustrated in [1].

Along with the Hash, a pointer to the user public key is provided in the model. (See the examples)

Uses

This methodology can be used, obviously, to create filters based on the identity of the signer. This methodology has also been successfully used for providing capabilities of "information revision" in RDFGrowth. In short, once a MSG has been signed, the hash can be used as Inverse Functional Property, that is, as way to univocally identify the MSG itself. This in turn can be used in a subsequent MSG to indicate the one that it substitutes. Given that the paternity of this subsequent MSG can be verified to be identical, the client can safely perform the information update, no matter where it received the update patch from.

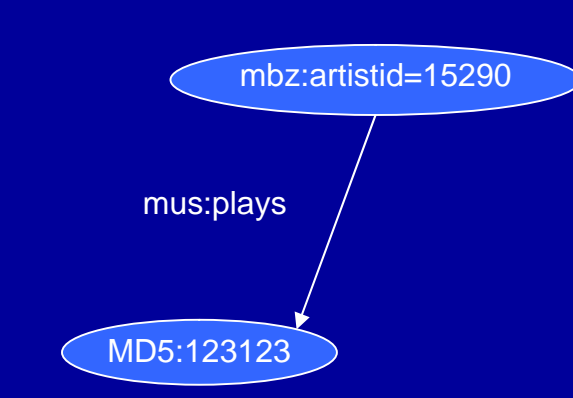
Issues

- 1) Since this methodology uses reifications as a way to attach the signature to the MSGs, it is subject to shortcomings typical of this standard RDF construct. In particular, care should be used when using this proposed method in OWL FULL reasoners as the owl:sameAs property might cause substitutions inside MSGs. RDFS inference presents similar problems, as new triples resulting from schema entailments could be automatically added by the RDFS triplestore involving blank nodes (thus potentially invalidating the signature).
- 2) By MSG definition and RDF Semantics, the structure of existing MSGs will not be affected by insertion of new ones. While this property enables our RDF digital signature schema, care must be provided not to insert an MSG twice, as it would result in a duplication in the database. A possible solution involves the use of a MSG identity operator. Such operator would, in a sense, be coherent with RDF Semantics according to which a graph statements are never repeated.

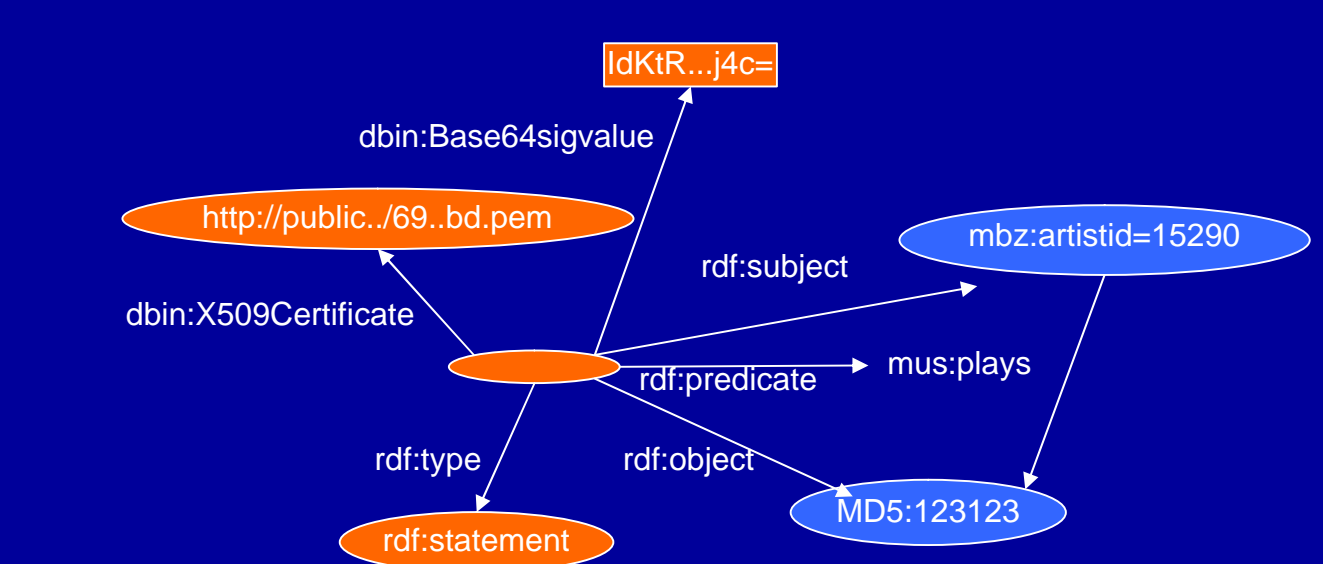
Conclusions

This methodology was developed to provide trust and information revocation procedure in DBin, a Semantic Web P2P platform [7]. In DBin, the issues highlighted above are overcome by using a "pipeline" of triple stores which somehow resembles the Semantic Web Tower: RDF information is stored and exchanged untouched into a main "raw" repository. Importing and processing information from the main one, satellite repositories can then be created supporting higher level reasoning as needed. An implementation of this method-

Simple statement (Trivial MSG)

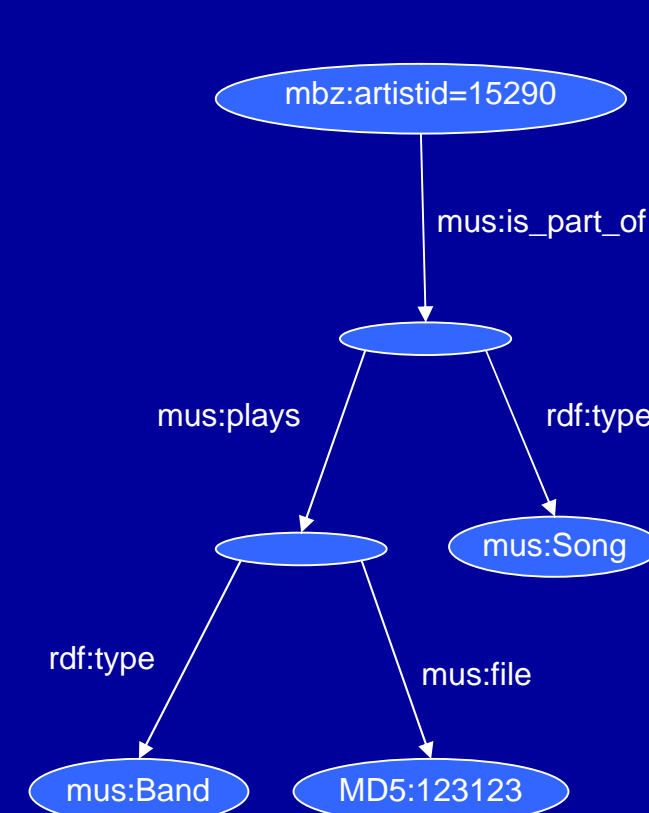


Signed Statement

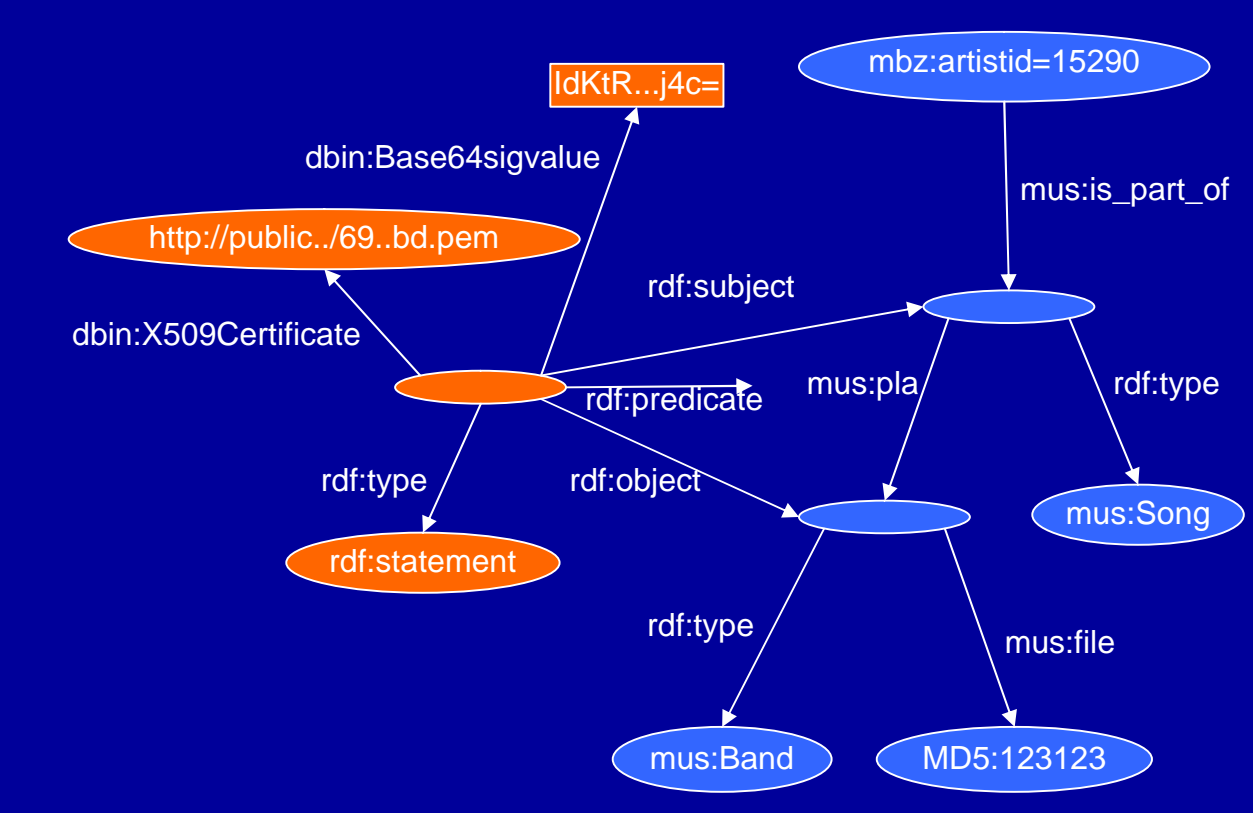


Signing single triples is possible but has a large overhead in terms of triples

MSG Unit



Signed MSG



Larger MSG can be signed by reifying a single triple, this lowers the relative overhead. In the "DBin" P2P application, typical MSG size is approximately 20 triples, thus signing overhead is reduced to approximately 25%.

7. References

- [1] J.Carroll, "Signing RDF graphs", HP technical report 2003
- [2] J.Carroll, C. Bizer, P. Hayes, P. Stickler, "Named Graphs, Provenance and Trust", HP technical report 2004
- [3] E. Dumbill, "Signign FOAF files" <http://usefulinc.com/foaf/signingFoafFiles> personal communication
- [4] RDF Semantics, W3C Recommendation, 2004
- [5] G. Tummarello, C. Morbidoni, J. Petersson, P. Puliti, F. Piazza, "RDFGrowth, a P2P annotation exchange algorithm for scalable Semantic Web applications", P2PKM, Boston 2004
- [6] P.Stickler URIQA The URI Query Agent Model, NOKIA 2003
- [7] Tummarello G., Morbidoni C., P. Puliti, F. Piazza, "The DBin Semantic Web platform: an overview", WWW2005 Workshop on The Semantic Computing Initiative (SeC 2005) www.DBin.org