Extensible markup language keywords search based on security access control

Meijuan Wang*

College of Computer Science and Technology,
Nanjing University of Aeronautics and Astronautics,
Nanjing 211106, China
and
College of Command Information Systems,
The Army Engineering University of PLA,
Nanjing 210007, China
Email: wangmeijuan1984@hotmail.com
*Corresponding author

Jian Wang and Kejun Guo

College of Computer Science and Technology,
Nanjing University of Aeronautics and Astronautics,
Nanjing 211106, China
Email: wangjian@nuaa.edu.cn
Email: guokejun@nuaa.edu.cn

Abstract: With increasing rate of storing and sharing information in the cloud by the users, data storage brings new challenges to the Extensible Markup Language (XML) database in big data environments. The efficient retrieval of data with protection and privacy issues for accessing mass data in the cloud is more and more important. Most of existing research about XML data query and retrieval focuses on efficiency or establishing the index, and so on. However, these methods or algorithms do not take into account the data and data structure for their own safety issues. Furthermore, traditional access control rules read XML document node in a dynamic environment, relevant dynamic query-based keyword research data security and privacy protection requirements are not many. In order to improve the search efficiency with security condition, this paper examines how to generate the sub-tree of matching keywords that the user can access by the access control rules for the user’s role. The corresponding algorithm is proposed to achieve safe and efficient keywords search.

Keywords: XML access control; keywords search; SLCA.


Biographical notes: Meijuan Wang is now a doctoral candidate at the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. She obtained her bachelor degree in 2004 at Anqing Normal University and master degree in 2007 at Nanjing Normal University. Now she is a Lecturer at PLA University of Science and Technology, and her major research areas are XML security storage and network access control.

Jian Wang is a Professor, PhD Supervisor in the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. He received a PhD degree in the Department of Computer Science and Technology, Nanjing University, China, in 1998. He was the postdoctoral fellow of the University of Tokyo in 2001–2003. Since 2004, he has been working in the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. He is a specialised Committee Director in Jiangsu province network and distributed computing, the chairman of the executive committee of Jiangsu province workstation in Chinese Association for Cryptologic Research, and his major research areas are key management, cryptographic protocol, privacy protection and security evaluation.

Kejun Guo is now a doctoral candidate at the College of Computer Science and technology, Nanjing University of Aeronautics and Astronautics, China. He obtained his bachelor degree in 2010 and master degree in 2013 at NUAA. He is interested in the areas of network security and attack detection.
1 Introduction

With the mounting demand for cloud computing (Vouk, 2008), data storage expansion from the TB level to the PB level far beyond the Moore’s law. The cloud data storage on heterogeneous cloud data access control and query has higher requirements. Semi-structured Extensible Markup Language (XML) data model not only provides network data exchange, but also can provide data management model exchange and dispose by internet according to the user’s needs.

The advantages of XML data in cloud computing environments become gradually more significant. With the increase of XML data and development of related industry standards Document Type Definition (DTD), XML database has becomes the important component of data representation, data exchange and data storage on the internet. Safe and effective use of XML data is more and more important. XML query is one of the vital research areas of data management, and now most of the results focus on efficiency (Li and Moon, 2001), accuracy, sorting the return results, establishing the index, and so on. In particular, the algorithm Smallest and Lowest Common Ancestor (SLCA) can well locate the keywords target in hierarchy structure. However, these methods or algorithms do not take into account the data and data structure for their own safety issues.

As a stand-alone database storage mode instead of simple data exchange mode, the user’s demand for XML database storage mode continues to increase. Although XML databases also have to introduce a view similar to traditional relational database theory, traditional access control rules for accessing XML data through requests to the server in dynamic cloud environments. There are some standard specifications on XML data access security, and also a series of user-defined security models, such as Role-Based policies Access Control (RBAC) and some other standardised formal description model.

When processing XML document node dynamically, a query engine judges one node to be accessed or not continually according to each access control protocol. These achievements address the specific access rules optimisation and access issues such as conflict resolution, and query-based keyword research data security and privacy protection requirements are not many. If a group of keywords to be retrieved exists in multiple SLCA sub-trees (Xu and Papakonstantinou, 2005), frequently repeated judgement would seriously increase the XML XPath query engine load, which is burden to the query engines for XML documents.

The existing research on the combination of both problems shows an unsatisfactory situation in time efficiency.

In this paper, comparing existing achievements and taking advantage of denial priority principle, we propose the XML data keywords search optimisation project, which is based on the view of roles. Corresponding algorithm New Secure SLCA (NSSLCA) for optimising keywords search in dynamic environment is given, which has a certain significance for data security effective management. Finally, we conduct an experimental study of our secure query evaluation techniques using three data sets from different source. Besides security ability, the experimental results show that our keywords search algorithm NSSLCA is more effective than existing approaches in both space and time performance.

2 Literature review

2.1 XML keywords query

XML query is one of the imperative research fields of data management and data application. With the development of XML database, XML data query and retrieval focus on efficiency, accuracy, sorting the return results, establishing the index, and so on. One of the most classic methods existing related to XML keywords search is the algorithm SLCA which returns the smallest and least common ancestor. Putting an XML document as a tree, the SLCA method returns a set of its smallest answer as sub-tree, which is defined as containing all the keywords and all its sub-trees can’t contain all the keywords. The root of this sub-tree is called SLCA.

There are also some other representative references. Based on PageRank algorithm, XRANK (Guo et al., 2003) first put forward ElemRank Algorithm which can determine the value of the element, and then determine the position of the element in the document coding through Dewey to establish the inverted index. It can not only save storage space but also improve retrieval efficiency, and can be naturally extend to hyperlink search engine query for HTML and XML mixed content. XSEarch (Cohen et al., 2003) is also an excellent technique of keyword search and supporting semantic query. Before the search, different from the XRANK, XSEarch not only requires the user to provide a given keyword but also returns results set Tag, which increases the complexity. But more importantly, it has provided a guarantee for the introduction of semantics.

Now, although there are some improved research on the combination of specific issues and the classic method, keyword search algorithm still pays more attention on optimising the value of information retrieval efficiency and Results Sort respect hierarchy. Taking into account the complex application environments safety and efficacy data, the algorithm also has a certain lack of expansion.

2.2 XML access control

2.2.1 XML access control protocol

The standard definition of Extensible Access Control Markup Language (XACML) is creating a flexible, system-independent, structure clear approach to express the access control rules. Compared with simple permission and refusal, it can achieve a more detailed access control. The advantages of XACML are its powerful function, versatility and openness, while it brings a negative side. Its system architecture is too complex. It is very difficult to achieve because it usually requires the combination of Security Assertion Markup Language (SAML), XML Key Management Specification (XKMS), XML Digital Signature (XML Signature) and XML Encryption. So most of the researches still rely on traditional
RBAC model, security content of access control mainly takes individual rights as the centre to carry out, most of them improve by role-based permissions combined with XML tree structure.

A more representative access control scheme is reference (Gavrila and Barkley 1998). Taking full advantage of XML Schema and XPath and instance-based (XML document) access control model, the researchers make the granularity of access control can control any node of the document tree. And they gave a formal description of security rules with seven tuples: ‘subject, target, path, privilege, access, propagation, priority’.

The following research on access control protocol usually uses simple three tuples (Subject, +/–Action, Object) (Mutata 2006). Subject represents an access subject; it’s usually a role’s user ID. Object represents the object of rule processing, usually using XPath to locate. Action represents a behaviour, and defines + as allowing accessing, – as refusing.

In addition, there are many studies (Emami and Zokaei 2005; Fan et al., 2004; Fundulaki and Marx, 2004) using different approaches to solve data access control. Some authors use four tuples Schema Role Access Control Policy (SRACP) (Li et al., 2011), but this kind of expression hasn’t been adopted extensively.

The research of reference (Parmar and Shi, 2002) is based on multi-document association, and effectively solves the slow accessing problem of XML document processing. The research of reference (Qi et al., 2005) is based on the form of three tuples to map the rule functions, combined with the evaluation result returned by the rule functions, and calculate the verdict. And the reference (Li et al., 2010) is based on the attribute to discuss the relationship between access request, attribute authority, strategy and determine process, and it gives the condition of terminating the access control decision process. But currently there is no unified optimisation model to solve the problem of the standard of the privacy protection in dynamic network data expansion.

2.2.2 XPath
Regardless of the formal description of access control, most of the results use XPath to locate Object, this paper also intends to follow. There are two reasons for using XPath to describe access control protocols:

1. XPath provides a variety of ways to reach the minimum unit of XML document structure, such as an element, an attribute, a text node, or a comment node. In this way, it allows policy-makers to write a strategy in a flexible way (e.g. allowing access to a certain element node, but not its corresponding attribute), in order to achieve the purpose of fine-grained access control.

2. For access control rules with conditional decision, accessing a certain nodes in XML document is often affected by a decision value. XPath can provide necessary restrictions by assertion expressions.

For example, we give an XML document tree with Dewey code in Figure 1. The XML Document Tree T1 is portion of information about the school curriculum and student elective.

In Figure 1, the constraint that ‘teachers’ salaries are not visible to students’ can be described as Rule R1.

1. $R_1$ (Role:STUDENT,–r, school/class/teacher/tsalary).

Although most access control rules are given based on role, there are constraints associated with the direct access to the main subject. The constraint that ‘the current login teacher/student can access the information of his own teaching/studying classes only’ can be described as Rules R2/R3. The constraint that ‘the current login student can’t access other students’ score’ can be described as Rule R4.

2. $R_2$ (Role:TEACHER,+R,/school/class[sname=#loginID]).

3. $R_3$ (Role:STUDENT,+R,/school/class[sname=#loginID]).

4. $R_4$ (Role:STUDENT,–r,school/class/students/student/score[sn ame!=#loginID]).

2.2.3 Deny priority
In many existing access model, we usually use ‘deny priority’ principle for overlapping access control rules.

1. An access control rule uses +R/–R to indicate spreading down through XML document structure by XPath, and uses +/-r indicates only applying to specified node but not spreading.

2. A refusal rule on a node overturns all permissible rules on the node and its sub-tree.

3. If there is no rule acting on a node, the refusal ‘–’ rule should be used on the node.

Figure 1 XML document Tree T1
3 Improvement of keyword search under access control

For XML T1 in Figure 1, taking into account the Rules R1 and R2, Rule R4 acts first on the element ‘score’ under the other student’s sub-tree. Assuming that the login student is Tom, R4 acts first on the node score (0.1.3.1.3) under the node student (0.1.3.0).

Although the deny principle can effectively protect the important data not to be access, it may cause over protection, which affects the user’s search results. If used improperly, it may even lead to unacceptable data access efficiency of the user.

More importantly, the deny priority principle will lead to the user unable to access the sub-tree data which may be not privacy data. We must address this issue when considering access control and privacy (Xiao and Tao 2006; Zhang and Zhao, 2005).

2.3 XML security query strategy

Most of the existing literatures about XML data security query use the post-processing scheme that query after screening, or match associated access control rules in the query process. They all meet the security requirements of data at the cost of query efficiency. There is almost no efficiency can be assessed in the latter method.

There are some achievements (Kudo and Hada 2000; Li et al., 2007; Liu and Chen, 2007; Sun et al., 2007; Lin et al., 2014) on the keywords search under the secure access control in XML document. The reference (Li et al., 2010) selects security attributes in the keyword search results, and combines XML keyword search scheme with the security view of access control. The idea of the algorithm has an intuitive description and a certain degree of desirability. However, the time complexity of the algorithm still has room for improvement. The reference (Li et al., 2011) further optimises query SLCA algorithms by establishing access to control rules index. This kind of literature separates data access security rule and keyword search scheme, there is obvious time redundancy in the actual operation.

In addition to the complexity of the optimisation algorithm and the effectiveness of the results, there is also another class of security query strategy literature, which is the privacy policy of compressed storage. The compression strategy of tree instance information that reference (Wu and Tang, 2011) proposed is: firstly, convert it to an instance information tree according to the rules of the XML document. Then, extract the global strategy tree according to the information of the instance information tree. Finally, react to the instance information tree. This method has strong pertinence and it can effectively compress the access control rules of XML, so as to optimise the efficiency of keyword search.

3.1 Structural frame view

Providing access control can be optimised during compiling, or be checked during running time (Gavrila and Barkley 1998).

3.1.1 Role

The concept of the Role exists very early. Role-based access control RBAC is a classic study of the results of safety issues. The plurality of subjects corresponding to a role may exist same access control structural frame view under access control rules.

Although an XML database may contain a lot of user entities, the usual role classification is limited. For example, the document environment given by this paper can usually be divided into three kinds: TEACHER, STUDENT and ADMIN.

3.1.2 DTD view

The security view framework based on the role got in static state can avoid multiple accesses, judgement and masking operation of a repeat path for the complete data files, so it can improve efficiency to a certain extent.

When accessing, different roles get different visual frameworks. As users of XML Document T1 in Figure 1 for example, the different roles ADMIN/TEACHER/STUDENT got the different visual XML documents through access control rules on DTD before the visit, as shown in Figure 2.

For example, under the effect of Rule R1, teachers’ salaries are not visible to the Role STUDENT. In other words, for access control rules do not contain assertions like Rule R1, we can prune sensitive data directly in the structural framework, and obtained the security framework view for a certain Role like STUDENT.
The operation of DTD View is not complex. If we use the formal described access control rules, when we scan rules we can get the secure DTD View in complexity $O(\cdot)$. Figure 2 The subject view

3.2 Subject deny priority

In access control constraints, the ‘deny priority’ principle can protect the fine-grained unit well to refuse the access to data privacy protection. And by $+R/-R$ it realises whether rules along the propagation path by XPath. These are the basic method for security target in data query operation. However, in view of the poor design of access control rules in the constraints of deny precedence may lead to low efficiency. This paper applies ‘deny’ priority rules on the subject access rules.

A STUDENT role for an XML database access protocol may contain multiple rules like R1, R3, R4. Before keywords searching, considering the necessary security, we have to firstly deal with these rules one by one. In order to improve the operating efficiency can be considered from the perspective of the rule processing order.

In the case of user’s information contained in the assertions, firstly to process these rules is the consideration of our algorithm. When accessing course resource that not the login users' elective courses, Rule R3 gives priority to effect on the sub-tree of ‘course’ node as root. When accessing another student’s score of the same elective course, Rule R3 gives priority to effect on the ‘score’ leaf node in the sub-tree of ‘student’ node as root which does not belong to the login user. For the instance, executing R3 and R4 can be much more effective than executing R4 and R3. Importantly, the scale of sub-tree pruned by Rule R3 is much greater than R4.

Therefore, considering the authority of the role of login users and entity in the context, the hierarchy of XML data and the corresponding structure of model and instance, data structure association and semantic association still use deny priority principle in the transfer process of access control rules pretreatment combined with the context, which can ensure the reasonable optimisation of the data in the following operation.

This paper puts forward ‘subject deny priority’: firstly deal with the data of subject information of the role in the assertion. Then deal with relevant rules according to the order of prohibit access path (the lower the access path hierarchy, the higher the priority) and ‘deny priority’ principle, prune all the inaccessible node and path of user subject that logged in currently.

4 Algorithm and experiment

Considering the role of the network context has a great impact on the use of the data. This section combines static and dynamic states, makes full use of the ‘subject negative priority’ strategy to optimise the query efficiency of SLCA algorithm in dynamic environment.

4.1 Algorithm

The proposed NSSLCA algorithm is listed in Figure 3.

Figure 3 The NSSLCA algorithm

```
Algorithm N(DTD D, XML T, keywords W, Rules R)
In: < D, T, W, R>, #loginID
OUT: n-set
//operation on DTD D
for each $t \in R$ do
    if assertion($t$) = NULL then
        if access(xpath $p$) = "deny" then
            val(xpath $p$) = 'N'
            shield-pruning(xpath $p$)
        else Rewrite $t$ into an inverse proposition
    end if
end if
end for
return D

// mapping the D’ to T’
// operation on XML T’
for each $t \in R$ do
    if assertion($t$) = #loginID then
        shield-tree(access(xpath $p$) = "deny"))
    end if
end for
for each $t \in R$ do
    if assertion($t$) then
        if access(xpath $p$) = "deny" then
            val(xpath $p$) = 'N'
            shield-pruning(xpath $p$)
        end if
    end if
end for
return T’
// operation on XML T’
return SLCA(T’)
```
Briefly, for different login user, get a different range of accessible.

Step 1 Static pretreatment

i View processing: Apply rules processing for access control rules which without assertion rules based on XML framework (DTD).

Find the access control rules of which does not contain assertion that according to the current role, mark the accessible attribute of XPath which is not allowed to assess node as ‘N’ and shield pruning. Then return the security framework view.

ii Rule Processing: Scan access rule.

Rewrite the permission access rule that contains assertions into an inverse proposition. For example, rewrite R3 into (Role:STUDENT,-r,//courses[sname!=#loginID]), in order to firstly unified consider deny priority principle in dynamic environment.

Step 2 Dynamic optimisation

i ‘Subject deny priority’ processing.

According to the ‘subject deny priority’ definition given by Section 3.2, firstly choose to deal with application rules of which assertion contains user-related rules based on XML document tree and shield the inaccessible node which does not meet #loginID attributes, which is usually regarded as sub-tree level and can execute pruning operation. This step can greatly simplify the size of XML document in dynamic environment.

ii Scanning processing.

Scan the rule which contains assertion, locate the inaccessible nodes in XML document and prune it. This step executes after security view mapping and ‘subject deny’ in the document tree, so the scale of object reduces much.

Step 3 Return the secure SLCA set

After operating step 1 and 2, go on search the keywords set W and return the secure SLCA set.

4.2 Algorithm analysis and example

All above, when we deal security keywords search on XML document, first step we deal XML structure framework by DTD pruning, then the size of the corresponding XML document tree will be relatively small. In the dynamic step after static processing, we first shield the sub-tree nodes which are not meeting #loginID property to effectively reduce the query matching number of retrieval. Finally, the result is calculated in secure accessible nodes, which meet the requirements of access control and keywords.

Ignoring the time cost of tree traversal, time cost of the algorithm we present is mainly related to the size of access control strategies, and its time complexity is $O(n)$. For a limited XML document tree, the algorithm will ensure convergence.

In case of the example in Section 2.2 and the Rules Set R in Section 2.3, assuming that the login user is a student ‘Joe’.

Before executing Algorithm SLCA, the model of XML document tree has been updated as Figure 4.

Figure 4 XML document after optimisation algorithm

![XML document diagram]

By comparison, we get that the XML document tree can be certain simplified. This algorithm is mainly based on reducing the cost of XPath and Node matching calculation to improve efficiency greatly, especially when the keywords contained in the sub-tree denied by the assertion condition not meeting the rules of access control. Considering the user ‘Joe’ has only one Course ‘Operating System’, and the corresponding information he can see was only contained in the sub-tree class (0.1). With the rule RX, we can prune the sub-tree class (0.0), class (0.2), etc. The keywords search is only on the visible nodes in Figure 4, the result final returned is security information root set {student (0.1.3)}. And if we first use keyword search and then screening filter access control conditions, whereby visible cases may not be effective security information root node.

Research about improving algorithm by keyword semantic association will be a focus of future work.

4.3 Experiment

In order to verify an XML-based keyword search algorithm NSSLCA access control effectiveness of the proposed, we respectively complete the accuracy of the algorithm under NSSLCA stack implementation and secure operation after SLCA. We pay attention to its query certain access control constraints and efficiency differences based on keywords. Configure the test environment for: processor is Intel core i5 2.50GHz, memory is 4G, the code is Java language, XML parsing method is DOM, and the compiler is in Eclipse.

In the platform based on the hardware and software environment, we use three data sets for testing. The processing time is measured from our own student information data collection, based on software tool XMLSpy custom data and actual data from DBLP set sizes of T1(25M), T2(3M), and T3(78M).

Figure 5 shows the result of our experiment. The horizontal axis represents the number of keywords of the query, the vertical axis represents the time required to execute different algorithms. Owing to space limitation, we do not explicitly list the test queries.
As shown in Figure 5, we can see that our new approach is more efficient than the previous work in most cases. According to experimental results, in the situation of same number of keywords and the security conditions in the rule to restrict access to the login role assertions related entities exist, NSSLCA algorithm has a very significant efficiency than the original method.

As can be seen from the results, the effect of optimisation makes quite a big difference. Refer to Figure 5 (b), the specific time efficiency doesn’t depend on the number of keywords, but rather on the number of conditions related to the semantics of deny.

5 Conclusion

XML Data Query is the one of the most significant parts of XML data. As XML database increasingly becomes the main form of cloud storage, users focus on Data-Access and Query-Efficiency. Moreover, the research is to use data securely and efficiently with a lot of access control rules is significant.

In this paper, we classify the rules of security access control base on Role, and formally present the static security (DTD view with ROLE) and dynamic security (XML Tree with SUBJECT-deny-priority). The result of return is the security SLCA nodes set, which can satisfy the demands of both access control and keyword query. And then, we present the theoretical analysis and experiments comparison to verify the effectiveness of our algorithm.

With the development of XML database, there are still some limitations of DTD described constraint. The XML Schema which integrates access control rule not only needs more standardised formal description, but also needs to consider the grammar relevancy of user-defined node. The further extension of algorithm analysis will be the following research.

Indeed, the role itself contains semanteme, then we have to optimise the formalised storage considering the semantic constraints. When the query-keywords distribute in different sub-tree and semantic associated, the returned SLCA nodes set may be much more than predicted. When the query-keywords contained in a sub-tree but semantic irrelevant, the returned SLCA nodes set may be not what we want, as the security information node \{student (0.1.3)\} in this paper. So, the search of multiple semantic associated keywords in different sub-trees will be the significant research in future.

Acknowledgements

This work was supported by Funding of Jiangsu Innovation Program for Graduate Education (the Fundamental Research Funds for the Central Universities) KYLX_0285, the Natural Science Foundation of Jiangsu Province (BK20150758), the Colleges and Universities young talents fund provincial key projects of Anhui Province (2013SQRL131ZD), and the pre-study fund of PLA University of Science and Technology.

References


