The application of XML-based Data Exchanging Technology in The Computer Forensics Field

Qin Liu

Department of Information Science and Technology, East China University of Political Science and Law, Shanghai 200042, China

Keywords: Data exchanging, XML, Relational database, Computer forensics

Abstract. XML has become the standard of data representation in the science and business application and standard of data exchanging among applications in Web. Driven by the demand and technology, various database vendors extend the function of XML. This article introduces an application system in the computer forensics field which is developed by taking advantage of XML extended function provided by commercial DBMS. First of all, use XML format as the standard of middle data exchanging in the course of evidence examination and then form the evidence chaining through comparison and eventually transform it into data in the HTML format and submit them to the court.

Introduction

With the universal popularization of computer and network, various attacks and damages to the computer and network also increase day by day. The computer is increasingly occurring in the criminal activities, and the court cases related to the computer also continuously occur and the electronic evidence existing in the computer and the relevant peripheral equipment has also become one of the new evidence in litigation[1]. In general, parts of the evidences are stored in the relational database of different types and other parts of the evidences are stored in the web pages. Since the structure and representation form of data are different, the data cannot mutually interact with each other. In order to obtain effective evidence chaining, the investigators often need to spend plenty of time finding and matching the evidence mechanically. Therefore, researching the data exchanging between heterogeneous data and applying it into the computer forensics field is becoming one of the indispensable investigation means. XML has become the factual standard of data representation in science and business application and data exchanging among applications[2]. Comparing with HTML, it has the characteristics of expandability, flexibility, self-describing and supporting multiple codes and the main relational database Management Systems (RDBMS) have all carried out the extension of XML support. Therefore, using XML as the intermediate form of data exchanging is a relatively good choice[3].

This article develops the application system of data exchanging technology in The Computer Forensics Field by taking advantage of XML extended function provided by DBMS.

XML Technology in RDBMS

The majority of DBMS have carried out the extension of XML support and promoted XML-Enabled products. If a product is to be called as XML-Enabled product, it must meet the following two most basic requirements: (1) can be saved as XML document in the database; (2) can generate XML document based on relational database, namely, the publication of XML. This article mainly introduces XML technology in DB2 and SQL Server.

IBM DB2 XML Extender units which are packed in DB2 and UDB8 are specially used to realize the overall or split storage of XML document and generate XML document from the existing relational database[4][5]. It has two available working patterns: (1) XML Column: use any type of user-defined types (XMLCLOB, XMLVARCHAR, XMLFILE) to store the whole XML document into certain column of the relational table; (2) XML Collections: permit the users to split XML document into table of DB2 or generate XML. Use a XML-format Data Access Definition (DAD) document to assign the mapping rules in the mapping process. In addition, some functions which also support SQL/XML [7] standard (formulating how to map between SQL and XML) mainly have three types: (1) functions that create elements and attributes, such as XMLELEMENT, XMLATTRIBUTES; (2) aggregating functions, such as XMLAGG; (3) assigning sub query of complicated nested structure. They can better support the publication of XML.

SQL Server 2005 provides users who are familiar with relational world with extended SQL sentences (FOR XML clauses and row set function OpenXML) so as to realize the storage and publication of XML and provide users who are familiar with XML with XML-based methods—creating XML view on the relational data[6].

System Design and Development

Overall Structure

Taking an economic dispute as example, merchant A is a retailer of books and merchant B is a wholesaler of books. Because of the accounting relations, merchant B accused merchant A of owing debts for years while merchant A argued that the products provided by merchant B were not in consistent with the demand and the requirements were still not met after many times returning or refunding. All transactions between the both parties were stored in the database or text files. Since both parties have cooperated for years, the accounting relations are complicated and purely relying on the technicians to verify one by one is time-consuming. Therefore, an evidence discovery system with XML as the intermediate exchanging data is developed. The system structure is shown as in Fig. 1, including several involved databases. Merchant A uses DB2 database and has several database servers and multiple database systems; merchant B uses SQL SERVER database. Since the DBMS they are using are different, the data mode is obviously different. This system uses XML as the standard of data exchanging to realize the data exchanging.

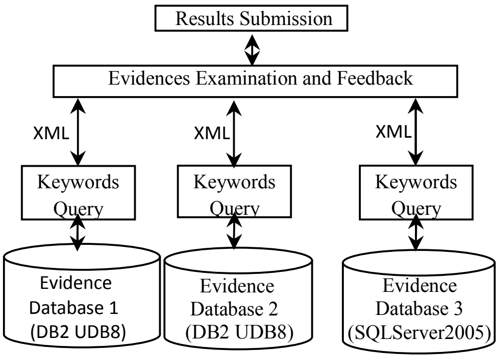
Results Submission
Evidences Examination and Feedback
XML
XML
XML
Keywords Query
Keywords Query
Keywords Query
Evidence Database 1 (DB2 UDB8)
Evidence Database 2 (DB2 UDB8)
Evidence Database 3 (SQLServer2005)

Fig. 1: System Structure

First of all, the investigators query out the relevant data from their own databases in accordance with the keywords and present it in uniform XML format; then, discover problems in accordance with the evidence examination of the first round and feedback the information in XML format, continue the query and repeat it for several times; eventually, find out the complete evidence chaining.

Key Technology in System Realization

This example is comprised of several interconnected PC machines which are database servers of the retailer and the wholesaler respectively.

```
<!ELEMENT order(opt_id, itemlist, total_price,fulfillment)>
<!ATTLIST order order_id CDATA #REQUIRED>
<!ATTLIST order order_date CDATA #REQUIRED>
<!ELEMENT opt_id(#PCDATA)>
<!ELEMENT itemlist(item+)>
<!ELEMENT item(title, author, publisher, pub_date?, quan)>
<!ELEMENT title(#PCDATA)>
<!ELEMENT author(#PCDATA)>
<!ELEMENT publisher(#PCDATA)>
<!ELEMENT pub_date(#PCDATA)>
<!ELEMENT quan(#PCDATA)>
<!ELEMENT total_price(#PCDATA)>
<!ELEMENT fullfillment(pay_mode, pay_date, acc_num, sum, opt?)>
<!ELEMENT pay_mode #PCDATA)>
<!ELEMENT pay_date #CDATA)>
<!ELEMENT acc_num #PCDATA)>
<!ELEMENT opt #PCDATA)>
```

Fig. 2: order.dtd

The retailer has two servers and the operation system is Windows2003; Web server software is Tomcat4.0; the database is DB2 UDB8 which is indicated as evidence database 1 and evidence database 2 in Fig. 1.

The wholesaler has only one server and the operation system is Windows2000 Server; Web server software is WebLogic7.0; database is SQL Server2005 which is indicated as evidence database 3 in Fig. 1.

Development language and tools: use XML, JSP, Servlet and so on to realize, and the development tools of JSP and Servlet is NetBeans8.

System of the Retailer

After the investigators inquiring the evidence database in accordance with the keywords and the system inquiring the relevant database in accordance with the keywords, the Retailer is invoked. Java program, which is mainly divided into the following several parts:

i) Checkout of the legality of keywords;

ii) XML generation: the query results are generated into XML in accordance with DTD as defined in Fig. 2. DTD in Fig. 2 indicates the detailed information, total amount, payment and performance and other information of the orders. The generation of XML takes advantage of DB2 XML Extender components (under pattern of XML Collections). First of all, compile DAD (shown in Fig. 3) of SQL_stmt style, then invoke the storage procedure db2xml.dxxGenXML() to generate XML in accordance with the relevant table. However, XML at this time is still stored in checkxml of result column in result table in the type of XMLVARCHAR. Finally, the following SQL sentences are needed to use to extract the required XML from the results.

```
rs=st.executeQuery
("select db2xml.extractVarchar(result.checkxml, '/orderlist') from result"+
"where db2xml.extractVarchar(result.checkxml, '/orderlist/order/@order_id')= "+id);
orderXML=rs.getString(1);
//string variable orderXML is used to save the order information and fund payment
```

```
…
<DAD>
   <validation>NO</validation>
   <Xcollection>
      <SQL_stmt>
SELECT orders.order_id, order_date, p.title, p.author, p.publisher, p.pub_date,
i.quan, f.total_price, f.pay, f.pay_date, f.acc_num
FROM orders, table(SELECT substr(char(timestamp(generate_unique())),16) as
i_id, order_id, itemid, quan  FROM orders_detail)  i , table(select id, pay,
total_price acc_num, pay_date from funds)  f
WHERE  order.order_id=i.order_id and i.itemid=f.id order by order_id, i_id, f.id
      </SQL_stmt>
      <prolog>?xml version="1.0"?</prolog>
      <doctype>!DOCTYPE order SYSTEM "C:\dtd\order.dtd"</doctype>
      <root_node>
      <element_node name="orderlist">
         <element_node name="order">
            <attribute_node name="order_id">
                 <column name="order_id" />
               </attribute_node>
               <attribute_node name="order_date">
                 <column name="order_date" />
               </attribute_node>
       <element_node name="itemlist">
               <element_node name="item" multi_occurrence="yes">
                 <element_node name="title">
                   <text_node>
                     <column name="title"/>
                   </text_node>
                 </element_node>
                           …
              </element_node><!--the end of item element-- >
               </element_node><!--the end of itemlist element-- >
          <element_node name="total_price">
                 <num_node>
               <column name="total_price"/>
              </num_node>
               </element_node>
       <element_node name="fullfillment">
               <element_node name="pay_mode">
                         <text_node>
                     <column name="pay"/>
                   </text_node>
                    </element_node>
                    …
             </element_node><!—the end of fulfillment-->
           </element_node><!--the end of order element-- >
    </element_node><!—the end of orderlist element-- >
      </root_node>
   </Xcollection>
</DAD>
```

Fig. 3: DAD of SQL_stmt style

iii) Sending the results: use XML data in string variable orderXML as the parameter and send the results to Servlet of the wholesaler in the form of POST request for handling. In this example, we uses the class of com.oreilly.servlet.HttpMessage realized by Jason Hunter to complete the task and the code is as follows:

```
URL url=new URL(SUPPLIER_URL);
 //use SUPLLIER_URL to save URL of the wholesaler
HttpMessage msg=new HttpMessage(url);
Properties p=new Properties();
p.put("order", orderXML);
 //confer order attributes to XML data
InputStream in=msg.sendPostMessage(p);
 //send data in attributes table p and receive the response data, if the sending is unsuccessful, then throw IOException,
//After capturing with method sendPostMessage, notify the administrator to send again.
```

System of the Wholesaler

Likewise, after the investigators inquiring the evidence database in accordance with the keywords and the system inquiring the relevant database in accordance with the keywords, the Wholesaler is invoked. Java program, which is mainly divided into the following several parts:

i) Checkout of the legality of keywords;

ii) The generation of XML: the query results are generated into XML in accordance with DTD as defined in Fig. 2. The database of the wholesaler is SQL Server2005, we use for xml clauses to generate XML document. The storage procedures are as follows:

```
CREATE PROCEDURE salexml @acc char(10) AS
      SELECT orders.order_id, order_date, p.title, p.author, p.publisher, p.pub_date, quan, f.total_price,
      f.pay_mode, f.pay_date, f.acc_num
      FROM orders, table(SELECT o_id order_id, itemid, quan  FROM orders_detail)  i , table(select id, pay
      pay_mode, price total_price acc_num, pay_date from funds)  f
      WHERE orders.customer=@acc and funds.acc_num=@acc  and  order.order_id=i.order_id and i.itemid=f.id
      order by order_id, i_id, f.id
       FOR XML AUTO, ELEMENTS
GO
```

iii) Handling of the results: use java program to directly invoke the storage procedures and the code is as follows:

```
 //fragment of Wholesaler.java program
StringBuffer salBuf=new StringBuffer
("<?xml version='1.0' encoding='GBK'?>");
String salXML;
try
{
      st_pro=conn.prepareCall
      ("{call salexml(?)}");
      st_pro.setInt(1,ac);
      //variable ac is the account of the merchant to be inquired
      rs.next();
      salXML=rs.getString(1);
      salBuf.append(productXML);
      salXML=salBuf.toString();
      //variable salXML is used to save the information on relevant accounts //paid and the corresponding
      commodities information
 }
catch (SQLException ex)
{     e.printStackTrace() ;
      System.exit(-1);
}
```

Evidence Examination and Feedback

The main job of evidence examination is account checking, enumerating the order information with clear accounts and also enumerating the order information with unclear accounts. After the information extraction of the above two aspects, all running accounts information is presented in relatively uniform XML document format and then the account No., date of payment, detailed commodities information are enumerated one by one in accordance with the account items by using XPath language. The specific steps are:

i) Inquire the remittance date and remittance amount of the retailer in accordance with the characteristics of some commodities (mainly including the book name, the author and the press).

ii) Inquire XML documents of the wholesaler in accordance with the date of accounts recording, amount and the characteristics of commodities. Here, considering the time difference between the remittance date and recording date, the query time can be set and adjusted. In addition, considering that the payment for multiple orders may be made together, all amounts which are larger than the remittance amount will be shown when inquiring the recorded amount.

iii) The investigators decide whether the accounts are clear and complete in accordance with the information inquired.

Results Submission

After the running accounts are clearly sorted out, the complete evidence chaining is needed to submit to the court. Although XML format is used as the standard of intermediate data exchanging in the system, it is difficult to some extent for the judges who do not graduate from computer specialty to understand if the evidence chaining is submitted in the form of XML. Therefore, XSLT (eXtensible Stylesheet Language) is used to transform the XML document into friendlier HTML document.

Conclusion

This article provides the specific application example of data exchanging technology in The Computer Forensics Field and enables the data stored in different databases to interact, thus finding out the evidence chaining and completing the investigation through using XML as the intermediate format. In the aspect of XML support, DB2 and SQL Server database management system have their own different pattern. Generally speaking, DB2 is relatively comprehensive and leaves relatively big space for the users so that users can define the storage or publish XML elements or attributes in the course and the correspondence of table or attributes in the relational database by them. SQL Server has a relatively fixed form but it is very convenient for the beginners to realize.

As the factual standard of data exchanging, XML is also widely applied in E-commerce, online education and other occasions that need data exchanging except being used in the computer forensics field [7].

Acknowledgements

The research work was supported by Youth Foundation of National Social Science under Grant No.13CFX054 and Youth Foundation of Chinese Ministry of Education under Grant No.11YJCZH175.

References

[1] Ding Li-ping, Wang Yong-ji Study on Relevant Law and Technology Issues about Computer Forensics [J]. Journal of Software, 2005, 16 (2):260-275.

[2] Subramaniam S., Haw Su-Cheng, Poo K. H. s-XML: An efficient mapping scheme to bridge XML and relational database [J]. Knowledge-Based System, 2012,27(3):369-380.

[3] Zhang Jian-mei, Tao Shi-qun, Reasoning about Structural Integrity Constraints for XML [J]. Chinese Journal of Computers, 2010, 33(12):2281-2290.

[4] Li Jing, Su Hou-qin. Comparing and Analysing Oracle XML DB and DB2 Pure XML on XML Document Storage and Retrieval Performance[J], 2012, 29(5):195-198

[5] Beyer K., Ozcan F., Saiprasad S.etc., DB2/XML: Designing for Evolution[C]. Proceedings of the 2005 ACM SIGMOD international conference on Management of data[C], New York, USA, 2005:948-952

[6] Rys M. XML and relational database management systems: inside Microsoft SQL Server2005[C], Proceedings of the 2005 ACM SIGMOD international conference on Management of data[C], New York, USA, 2005:958-962

[7] Mishra N., Chaturvedi S., Sharma A. K. etc., XML-Based Authentication to Handle SQL Injection[J], 2014,236(2):739-749