전자서명 시스템을 위한 XML 정규화 알고리즘 설계 및 구현

유윤식* · 이강찬** · 전종홍** · 이원석** · 정회경*
*배재대학교 컴퓨터공학과 · **한국 전자 통신 연구원

Design and Implement of Canonical XML Algorithm for Digital Signature System

Yoon-Sik Yoo* · Kang-Chan Lee** · Jonathan Jeon** · James Lee** · Hoe-Kyoung Jung*
*Dept. of Computer Engineering Paichai University · **ETRI
E-mail : ys5315@hanmail.net, {chan, hollobit, wslee}@etri.re.kr, hkjung@mail.pcu.ac.kr

요 약

오늘날 XML이 전자 상거래 시장에 널리 수용하여 사용되고 있다. 하지만 XML 문서는 논리적으로 동일한 의미를 물리적으로 여러 다른 형태로 나타낼 수 있는 표현의 자율성이 존재하기 때문에 XML 전자서명과 같은 물리적 형태로써 유효성을 판단하는 응용프로그램에서는 문제가 발생할 수 있다. 따라서 이런 문제점을 해결하기 위하여 W3C에서는 논리적으로 동일한 의미의 XML 문서를 물리적으로 동일하게 변환시키도록 XML 정규화 알고리즘을 제안하여 사용하도록 권고하고 있다.

이에 본 논문에서는 W3C에서 권고한 XML 정규화 알고리즘을 수행하는 시스템을 설계 및 구현함으로써, 좀 더 정교하고 정규화된 문서로 변형하여 W3C 표준을 따르는 다른 응용 시스템과의 상호 운용이 가능하다. 또한 웹 서비스를 위한 전자서명 시스템에서의 사용이 용이하며 웹 서비스 상호 운용성을 위한 XML 문서 교환 시 물리적 동일성이 요구되는 여러 시스템에서 다양한 방면으로 사용될 것으로 사료된다. 뿐만 아니라 국제적 인코딩 스킴과 국내 인코딩 스킴인 EUC-KR과의 변환기능을 추가함으로써 국내 실정에 맞는 XML 정규화 알고리즘이 될 것이며, 이는 국제적 상호 운용성 확보의 기반 기술이 될 것이다.

ABSTRACT

These days, XML is accepted and used to e-commerce market broadly. But by reason of XML document has autonomy of expression that can exist same form logically but several other forms physically, several problems can happen in application that judge effectiveness as physical form such as XML digital signature. Therefore, it is recommending to propose and use Canonical XML algorithm to change identical XML document physically equally logically in W3C to solve this problems.

We implemented system that run Canonical XML algorithm that suggested in W3C that can change to more elaborate regular document. Thus, interpretable with other application that takes W3C recommendation Also, as well as use in digital signature system for web service is useful, use in several system that physical identify is required when it exchanges XML document for web service interoperability are considered to be valuable. Moreover, Adding the transformation ability between universal encoding scheme and EUC-KR that is internal encoding scheme should be Canonical XML Algorithm that is suited to internal circumstances, and this should be a foundation technique of international interoperability confirmedness.

키워드
XML, 전자서명, 정규화, 상호 운용성

I. 서 론

최근 인터넷이 활성화되고 사용자가 급속도로 증가하는 추세에 맞추어 기존의 기업들이 현재 갖고 있던 상거래 시장에서 인터넷을 통한 시장 확장을 목표로 전자 상거래에 관심이 모아지고, 각 기업들은 독자적인 전자 상거래 시스템 구축에 많은 투자를 하였다. 이에 발맞추어 차세대 웹 문서인 XML(eXtensible Markup Language)[1]을 전자 상거래 시장에서 수용하게 되었으나 전자 상거래의 특성상 문서의 신뢰와 투명성을 요구하게 되었고, 요구사항을 해결하기 위하여 여러 업체들은 각각 자신들의 독자적인 전자서명을 사용하게 되었으나, 다른 시스템들과 상호 운용에 있어서 문제점이 발생하게 되었다. 이런 이유로 XML의 장점을 충분히 활용하지 못하고 있는 것을 느낀 XML 사용자들은 공개적으로 사용할 수 있는 XML 전자서명을 요구하게 되었고, 이에 맞추어 W3C(World Wide Web Consortium)에서 XML Signature 표준 명세[2]를 제안하고 사용하도록 권고하고 있다. 또한 XML 정규화 명세[3]를 제정하여 이와 더불어 사용하도록 권고하고 있다.

이에 본 논문에서는 일반 XML 문서를 받아들여 XML 정규화 명세에 따라 정형화된 XML 문서로 변환시키는 XML 정규화 모듈을 설계 및 구현하였다. 이는 XML 전자서명뿐만 아니라 XML을 이용한 메시지 교환을 위한 응용분야에서 XML 문서의 물리적 동일성을 보장하는데 사용될 수 있다. 또한 UTF-8, UTF-16과 EUC-KR의 상호 변환 기능을 추가함으로써 국제적 상호 운용성을 보장할 수 있다.

II. 전자서명과 XML 정규화

1. 물리적 동일성에 따른 전자서명의 유효성

논리적 동일성이란 XML문서의 의미가 동일함을 뜻하고 물리적 동일성이란 공백문자를 포함한 모든 문자열이 같은 문서를 말한다. 즉, 논리적으로는 동일하지만 물리적으로 상이하다는 것은 XML 문서 표현의 자율성에 의하여 의미는 같지만 문자열이 다른 문서를 뜻한다. 다음 두 예는 논리적으로는 동일하지만 물리적으로 다른 구조를 가진다.

```
<name a ="1" b="2" c="3"></name>
```

```
<name c='3' b='2' a='1'/>
```

이러한 표현의 자율성은 XML 전자서명과 같은 문서의 물리적 구조를 중요시 하는 응용프로그램에서는 치명적 문제를 발생시킨다. 다음의 예는 위의 두 문서를 SHA-1 해쉬 알고리즘을 통하여 생성된 digest이다.

```
zIlvxonF84RYwgaWCt7hMOL3STo=
```

```
FqPFgYm5HkfoDEJnxJ6Lt/oqfZU=
```

두 digest값을 비교하여 그 값이 일치하여야 전자서명이 확인될 수 있기 때문에 위의 두 예는 전자서명이 깨지게 된다. 따라서 여러 가지 표현이 존재할 수 있는 XML 문서를 일관된 형식으로 변환시켜 물리적 동일함을 유지시키는 정규화 알고리즘이 필요하게 된 것이다. 정규화 알고리즘은 W3C Canonical XML 명세를 따른다.

2. XML 정규화

W3C에서는 XML 정규화 명세를 통하여 다음과 같은 규칙을 권고하고 있다.

- 속성값의 표준화
- 문자와 파싱된 엔터티 참조는 대체
- XML 선언과 DTD 제거
- 공백 엘리먼트는 시작 태그와 종료태그의 쌍으로 대체
- 문서 엘리먼트의 외부 공백과 시작 태그와 종료 태그에 있는 공백을 표준화
- 문자 내용의 모든 공백은 유지
- 속성값 구분자는 이중 따옴표로 대체
- 속성값과 문자 내용의 특수 문자들은 문자 참조로 대체
- 엘리먼트에서 불필요한 네임스페이스 제거
- 각 엘리먼트에 디폴트 속성 추가
- 각 엘리먼트와 네임스페이스는 사전적 순서에 의해 정렬
- 문서는 UTF-8로 인코딩
- 행 종료문자는 #xA로 대체

위의 규정들을 통해 응용 프로그램뿐만 아니라 사용자에 의해 변형 될 수 있는 의미적으로는 동일하지만 물리적으로 상이할 수 있는 문서를 정규화 할 수 있다.

III. XML 정규화 알고리즘 설계

W3C에서 정의한 XML 정규화 명세에 따르면 입력으로 두 가지 파라미터를 받아들이도록 되어 있다. 첫번째 파라미터로는 주석을 넣을 것인지 결정하는 주석처리 불리언(Boolean)값이고 두 번째 파라미터로는 XML 문서를 옥텟 스트림(Octet Stream)으로 받아들일 것인가, 아니면 노드 셑(Node Set)으로 받아들일 것인가를 결정하는 것이다. 따라서 본 모듈의 인스턴스(instance)를 생성할 때 주석처리 불리언값을 파라미터로 받아들이고 생성한 인스턴스의 직렬화 모듈을 호출할 때, 그 파라미터로 파일 또는 노드셋을 받아들이도록 설계하였다. 그림 1은 전체 처리 흐름도를 UML의

활동 다이어그램으로 표시한다.

그림 1과 같이 인스턴스를 생성할 때 첫 번째 파라미터로 받아들이는 값은 주석 처리 불리언이다. 이 값에 따라 결과 문서에 주석을 추가할 것인지 아닌지를 결정한다. 그 뒤 Write 클래스를 생성하고 그 파라미터로 변환될 문서의 인코딩 방식을 지정하여 다양한 인코딩 스킴(scheme)에 따라 문서를 변환시킬 수 있다. 기본 값으로 W3C에서 권고한 UTF-8이 지정되어있지만 EUC-KR을 지정하여 국내 실정에 맞는 XML문서로 변환 시킬 수 있다. 두 번째 파라미터로 받아들이는 값은 문서를 받아들이는 형식인데, 그 값이 문서 형식이면 바로 SerializeNode 모듈로 보내지고, 파일 형식으로 받아들였다면 문서 형식으로 변환한 뒤에 보내진다. 이 SerializeNode 모듈은 받아들인 노드의 형식과 5가지 노드의 형식(Document, Element, Text, Processing Instruction, Comment)을 비교하여 그 중 일치하는 노드에 따라 분기 하는 역할을 수행하고 이 중 Element노드에서는 그에 해당하는 Namespace와 Attribute노드를 처리한다. 분기된 노드는 그에 대응하는 처리 모듈로 보내지게 되고 각 처리 모듈은 DOM을 이용하여 원본XML 문서로부터 가져온 데이터와 해당 노드에 적절한 텍스트를 혼합하여 임의의 텍스트에 내려 쓰는 형식이다.

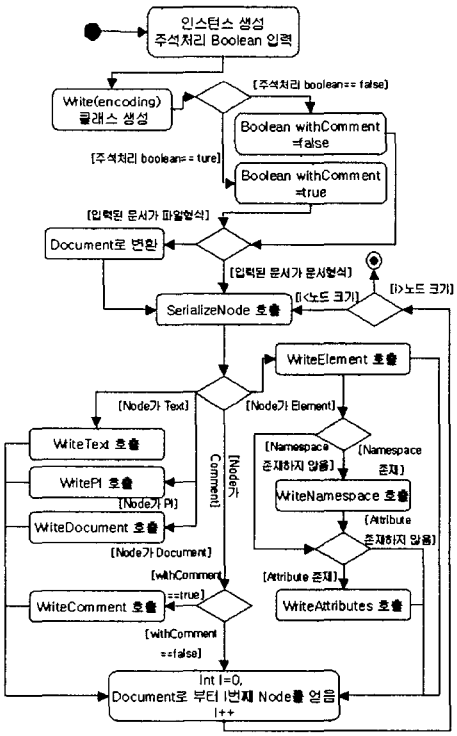

그림 1. 전체 시스템 구성도

IV. 구현

본 시스템은 IBM PC 호환 컴퓨터(Pentium IV 2.4G)에서 Windows 2000 운영체제 하에서 개발 도구로 JBuilder 8.0을 사용하였으며, 개발 언어로는 Java JDK 1.4를 사용하여 앞 장의 설계에 따라 구현하였다. 문서를 생성하고 DOM 객체를 생성하기 위한 XML 파서(Parser)로는 MetaStuff의 DOM4J[4]를 사용하였다.

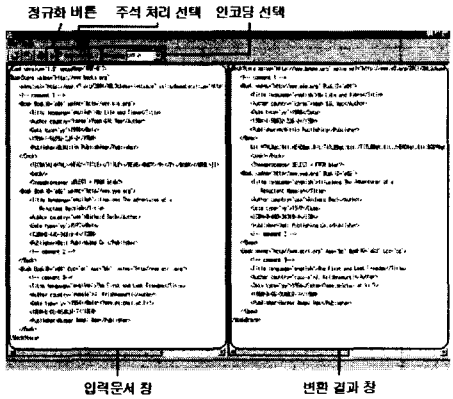

그림 2. 사용자 인터페이스

그림 2는 시험을 위한 사용자 인터페이스 화면이다. 인터페이스의 좌측 창에는 정규화 시킬 원본 XML 문서를 보여주고 그 상단의 도구막대의 정규화 버튼을 클릭하여 발생하는 이벤트로 정규화 모듈을 호출하여 문서를 정규화 한다. 그 정규화된 결과 화면을 우측 창에 보여줌으로써 입력 문서와 변환된 결과 문서를 비교하여 볼 수 있다. 또한 주석을 처리할 것인가에 대한 선택사항을 체크박스 형식으로 선택할 수 있도록 하였고 인코딩 형식을 선택할 수 있도록 하였다.

```
입력 문서
<?xml version="1.0" encoding="UTF-8"?>
<BookStore xmlns="http://www.books.org"
xmlns:xsi="http://www.w3.org/2001/XMLSchema instanc
e" xsi:schemaLocation="http://www.books.org
BookStore1.xsd">
    <!--This is Comment-->
    <Book BookID="a01" xmlns="http//www.mie.org">
      <Title language="english">My Life and
Times</Title>
      <Author country="korea">Gil Dong,
Hong</Author>
    </Book>
<![CDATA[<HTML><HEAD><TITLE></TITLE></HE
AD><BODY><P></P></BODY></HTML>]]>
<Book/>
<?nameprocessor SELECT * FROM blah?>
    <Book BookID="a03" Ccc="cc" Aaa="bb"
xmlns="http//www.etri.org">
      <Title language="english">The First and Last
Freedom</Title>
      <Author country="russia">J.
Krishnamurti</Author>
    </Book> </BookStore>
```

그림 3. 입력문서

변환된 문서
<BookStore xmlns="http://www.books.org" xmlns:xsi="http://www.w3.org/2001/XMLS chema instance"xsi: schemaLocation="http://www.books.org BookStore1.xsd">
<!--This is Comment-->
<Book xmlns="http//www.mie.org" BookID="a01">
<Title language="english">My Life and Times</Title>
<Author country="korea">Gil Dong, Hong</Author>
</Book>
<HTML><HEAD><TITLE></TITLE></HEAD><BODY><P>
</P></BODY></HTML>
<Book></Book>
<?nameprocessor SELECT * FROM blah?>
<Book xmlns="http//www.etri.org" Aaa="bb" BookID="a03" Ccc="cc">
<Title language="english">The First and Last Freedom</Title>
<Author country="russia">J. Krishnamurti</Author>
</Book> </BookStore>

그림 4. 변환된 문서

그림 3과 4는 입력문서와 본 알고리즘을 이용하여 정규화시킨 문서이며 지면 관계상 개행된 부분이 있다. 변환 결과를 살펴보면, XML선언부가 제거되었음을 볼 수 있고, 시작 태그 내의 네임스페이스 사이에 존재하는 불필요한 공백문자와 개행문자가 제거되었음을 볼 수 있다. 주석 노드의 경우에는 주석을 추가하도록 선택하여 결과 문서에는 주석이 포함되었음을 볼 수 있다. 그러나 선택을 해제하고 정규화 시킨다면 주석이 포함되지 않는다. CDATA 섹션 처리는 그 내부 값은 참조되는 엔터티 값으로 대체시켜야 한다는 명세에 따라 이 노드를 일반 텍스트와 같이 취급하여 텍스트 노드 처리에서 처리하여 해당하는 문자들이 대체되었음을 확인할 수 있다. 엘리먼트에 존재하는 속성들과 네임스페이스의 순서가 네임스페이스, 속성 순으로 정렬되고 여러 속성들은 알파벳순으로 정렬되었음을 볼 수 있다.

V. 결론 및 향후 연구방향

최근 인터넷이 활성화되고 이에 대한 관심이 높아짐에 따라 전자 상거래 시장에서는 XML을 수용하여 데이터를 교환하게 되었으나 이기종간 물리적으로 상이한 XML문서 구조를 가지고 있다면 XML 전자서명과 같이 물리적 형태로써 유효성 여부를 판단하는 응용프로그램 등에서는 치명적인 문제를 야기시킬 수 있다. 이러한 문제점의 해결을 위한 대안으로 XML 문서의 전자서명 보안기능을 제공하는 W3C의 권고안인 XML 정규화 변환 알고리즘이 존재하지만 해당 알고리즘을 곧바로 적용하는데 따른 관련 연구가 선행되어야 한다. 따라서 본 논문에서는 논리적으로는 동일하지만 물리적으로 상이할 수 있는 XML 문서를 물리적으로 동일하도록 재구성하여 이러한 문제를 해결하도록 하는 XML 정규화 시스템을 설계 및 구현하였다.

본 논문에서 제안한 시스템의 특징은 XML 정규화 표준 명세를 적용시킴으로써 다양한 XML 파서로 인한 상호 운용성 문제를 해결하였고 작성자에 따라 논리적으로는 동일하지만 물리적으로 상이할 수 있는 XML 문서를 동일한 물리적 구조를 가지도록 함으로써 좀 더 정교하고 정규화된 문서로 변형할 수 있다. 또한 본 시스템을 모듈화 하여 생성함으로써 다른 여러 응용프로그램에서도 이 모듈을 사용하여 정규화 시킬 수 있다. 따라서 XML 문서의 정규화를 통하여 전자서명 시스템에서의 사용이 용이할 뿐만 아니라, XML 문서 교환 시 물리적 동일성이 요구되는 많은 응용분야에서의 핵심 요소 기술로 사용될 수 있으리라 사료된다. 특히, 현재 사용되는 다양한 인코딩 스킴과 EUC-KR 인코딩 스킴과의 상호 변환기능을 제공함으로써 국내 실정에 맞는 기반 기술이 되어 국제적인 상호 운용성의 문제점을 극복할 수 있을 것이다.

향후 연구방향으로는 현재 웹 서비스에서 전송 프로토콜로 사용되는 SOAP(Simple Object Access Protocol)[5]을 정규화 시키기 위한 연구[6]가 진행되어야 하며, 또한 XML 전자서명 시스템과 연계하여 통합형 웹 서비스 보안 모델[7]에 관한 연구를 진행할 계획이다.

참고 문헌

[1] Extensible Markup Language 1.0, http://www.w3.org/TR/REC-xml

[2] XML Signature Syntax and Processing, http://www.w3.org/TR/2001/PR-xmldsig-core-20010820

[3] Canonical XML Version 1.0, http://www.w3.org/TR/xml-c14n

[4] DOM4J 1.4 API, http://www.dom4j.org/javadoc

[5] SOAP Version 1.2 Part 1: Messaging Framework, http://www.w3.org/TR/soap12 part1

[6] SOAP Version 1.2 Message Normalization, http://www.w3.org/TR/soap12-n11n

[7] Mark O'Neill, Web Services Security, McGraw-Hill, 2003