효과적인 접근제어 정책의 생성을 위한 XACML 정책 뷰어

최재호, Scott Uk-Jin Lee

한양대학교 컴퓨터공학과
경기도 안산시 상록구 사 3 동
jaeho34@hanyang.ac.kr, scottlee@hanyang.ac.kr

요약: IT 기기의 발전에 힘입어 자원의 양이 거대해지고 그 가치가 높아지는 시대가 되었다. 이러한 부가가치가 높고 거대한 자원을 올바르게 관리하기 위해서는 그에 맞는 적절한 정책이 생성되어야 하고 접근의 효율성 및 보안성을 높이기 위하여 생성된 정책에 따라 자원의 접근을 제어해야 한다. 따라서 올바른 접근제어 정책을 생성하기 위해 OASIS 에서는 XACML 을 표준으로 채택하였다. 하지만, 정책을 생성하는 과정에서 많은 어려움과 문제점이 발생한다. 특히, 정책의 부적절한 생성으로 인하여 충돌이 발생하고 이로 인하여 자원의 유출 및 의도하지 않는 접근을 허용하게 된다. 본 논문에서는 위와 같은 문제를 해결하기 위하여 효율적인 정책 생성을 위한 GUI 기반의 XACML 정책 뷰어를 개발하고자 한다. 이를 통하여 기존의 정책과 새로운 정책간의 직관적으로 비교, 분석이 가능하도록 한다.

핵심어: 접근제어, XACML, XACML 정책 뷰어

1. 서 론

최근 여러 IT 기기가 발전함에 따라, 인터넷의 접근성 및 사용량이 급속도로 증가하고 있는 추세이다. 이와 더불어 인터넷으로 공유되는 자원의 양 역시 급증하게 되었고 이러한 자원을 효율적으로 관리하는 방법이 화두가 되고 있다. 그러한 방법 중에 하나인 접근제어 (Access Control)는 클라우드 컴퓨팅 (Cloud Computing) 및 빅 데이터 (Big Data) 등과 같은 다양하고 방대한 양의 자원을 활용하는 서비스 분야에서 매우 중요한 기술이다. 특히, 클라우드 컴퓨팅과 같이 모든 사용자가 개개인의 자원에 접근하기 위한 권한이 존재하고, 이러한 권한에 따라 접근을 허용하는 환경에서는 세밀하고 유동적인 접근제어가 필수 요소이다. 이를 위하여 OASIS 국제표준위원회는 접근제어 권한을 적절히 표현하여 적용할 수 있는 정책 언어인 XACML (eXtensible Access Control Markup Language)을 표준으로 채택하였다 [1]. XACML 은 XML 에 기반을 둔 접근제어 정책 언어로 현재 대부분의 대규모 분산시스템에서 적절한 접근제어를 구현하기 위하여 사용되고 있다 [2].

XACML 은 다양한 분산시스템 환경에 적용 가능한 정책 생성을 자유롭게 허용하지만 이에 따른 기존의 정책과 생성된 정책간의 충돌이 발생할 수 있는 문제점이 존재한다. 이러한 충돌을 방지하기 위하여 정책 조합 알고리즘, 규칙 조합 알고리즘을 통해 새로운 정책과 규칙이 생성되지만 이는 정책 관리자가 의도하지 않은 새로운 정책으로 변질될 가능성이 있다. 무엇보다도 모든 정책을 정책 관리자가 직접 분석하지 않는 이상 변질된 정책을 발견 및 수정하기가 쉽지 않다. 이처럼 조합 알고리즘을 통해 만들어진 정책은 의도하지 않은 접근 권한의 부여를 가능하게 하고 이는 자원을 누출로 이어져 심각한 피해를 자초할 수 있다.

XACML 은 XML 에 기반을 둔 접근제어 정책 언어로써 세밀하고 세분화된 정책을 생성하기 위해서는 XACML 정책 구조에 관한 많은 지식이 요구된다. 특히, 클라우드 컴퓨팅과 같이 대규모 분산 시스템이 사용되는 환경에서는 자원의 관리를 위해서 필수적으로 정책이 생성되어야 한다. 이때, XACML 의 구조를 정확하게 인지하고 각 요소의 사용법과 생성 시 필요한 알고리즘을 모두 이해해야만 해당 환경에 맞는 정책을 생성할 수 있다. 이러한 점은 일반 사용자와 정책 관리자에게 많은 시간을 할애하게 하고 지식이 부족한 상황에서 정책을 생성하게 되면 기존에 존재하는 정책과 충돌이 발생하여 의도하지 않은 변질된 정책이 생성될 수도 있다. 그리고 정책을 생성할 당시에 기존의 정책과 비교, 분석하여 충돌여부를 확인하기 위해서는 직접 적으로 모든 정책을 들여다보고 판단해야 하는 어려움이 있다. 특히, 자원의 양이 증가함에 따라 정책의 수가 증가하고 복잡해지면 관리자가 정책을 직접 비교, 분석하기는 사실상 불가

능하다. 따라서, 정책 생성을 도와주는 에디터가 필요함은 물론, 기존의 정책과 새로운 정책을 비교, 분석할 수 있는 GUI 기반의 직관적이고 쉬운 사용자 인터페이스가 필요하다.

본 논문에서는 XACML 의 구조를 살펴보고 현존하는 XACML 정책 생성 도구의 복잡한 인터페이스의 문제점을 해결하는 직관적이고 쉬운 인터페이스를 만들고 효율성을 비교한 뒤, 이를 기반으로 한 도구를 개발하고자 한다.

2. XACML 정책의 구조와 예시

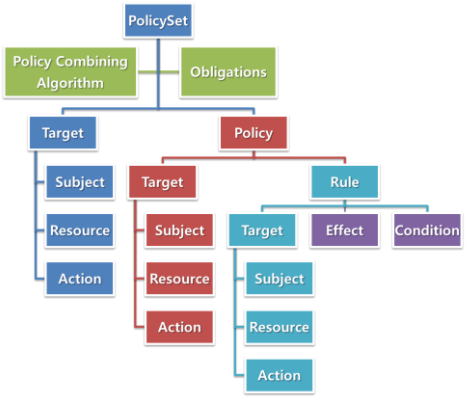

그림 1. XACML 정책 구조

XACML 정책은 그림 1 과 같은 구조를 가진다 [3]. 접근제어 정책이 기술되는 Policy 가 모여 집합을 이루고 각 Policy 의 결과를 종합하기 위한 정책 조합 알고리즘(Policy Combining Algorithm)과 정책을 판단할 때 지켜야 하는 정적인 제약조건의 정보가 저장된 Obligations 의 값을 이용하여 접근 권한 결과를 판단한다. 그리고 Policy 는 해당 정책의 Target 에 관한 정보와 정책의 접근 권한을 명시하는 Rule 로 구성이 되어 있다. 해당 Policy 의 Target 과 Rule 의 Target 을 바탕으로 접근을 허용(Permit) 또는 거부(Deny)하는 Effect 요소를 결정하고 Policy Combining Algorithm 과 Obligation 값을 이용하여

```
<Target>
  <AnyOf>
    <AllOf>
      <Match>
        <AttributeValue />
        <AttributeDesignator />
      </Match>
    </AllOf>
  </AnyOf>
</Target>
```

그림 2. XACML Target 의 구조

최종적인 접근 권한을 판단한다. 그리고 더욱 세밀한 접근제어를 구현하기 위해 Condition 요소에 논리연산을 지정한다면 Effect 요소와 조합하여 결과를 반환한다.

접근 권한을 결정하기 위하여 필요한 주체(Subject), 자원(Resource), 행동(Action), 환경(Environment) 요소로 구성된 Target 은 그림 2 와 같은 구조를 가지고 있다. 이 중에서 자원에 접근하기 위한 사용자의 요구사항과 정책을 비교할 때 사용되는 요소는 Match 이다. Target 은 여러 개의 Match 속 AttributeDesigner 와 AttributeValue 를 이용하여 주체, 자원, 행동, 환경 요소와 사용자 요구사항을 비교하여 결과를 도출하고 PolicySet 의 모든 Target 의 결과를 종합하여 접근권한을 결정한다.

그림 3. ABAC, RBAC 모델의 정책 예시

그림 3 은 요소 기반의 접근제어(ABAC) 모델의 정책과 역할 기반의 접근제어(RBAC) 모델의 정책 예이다 [4]. ABAC 모델은 주체, 자원, 행동, 환경 요소를 기반으로 정책을 생성하는 모델로 그림 4 의 윗부분을 보면 Target 과 Rule 이 Subject, Resource, Action, Environment 로 이루어져 있다. 이 예시는 자원인 Course.pdf 파일에 접근을 하려는 사용자에 관한 정책을 표시한 것이다. 그 중, Rule A 는 User A, B, C 가 어떤 시간에도 해당 파일을 다운로드 받을 수 있다는 것을 명시하고 Rule B 는 모든 사용자는 18:00~24:00 에는 해당 파일을 다운로드 할 수 없다는 것을 명시한다. 그림 4 의 아래 부분은 RBAC 모델로 역할에 따라 정책이 생성되고 각 정책에 Rule 이 적용되어 있다. 전체적인 PolicySet 의 Target 은 Tom 이라는 사람이고 그 사람의 역할인 Ph.D, TA, Student 에 따라 Rule 이 분류되어 있다. 현재 Tom 의 역할은 Ph.D 고 동시에 TA, Student 의 역할도 가지고 있다. 여기서 Rule 은 TA 는 MidTerm Grade.xlsx 라는 자원을 Edit 할 수 있고 Student 는 이를 할 수 없다는 것을 명시하고 있다. 직접적인 예

를 보더라도 XACML 의 지식이 부족한 경우 쉽게 알아보기 힘들고 어떤 요소가 어떻게 사용되는지 판별하기가 쉽지 않다.

이렇듯 복잡한 구조를 가진 XACML 의 정책을 생성하기 위해서 사용할 수 있는 사용자 인터페이스는 한정적이고 많은 전문지식을 요구한다. 이러한 이유로 정책 생성을 부주의하게 할 경우, 자원의 낭비 및 충돌 문제가 발생할 수 있고 이를 발견하고 수정하기 위한 효율적인 인터페이스가 필요하다.

3. 기존의 정책 생성 도구의 인터페이스

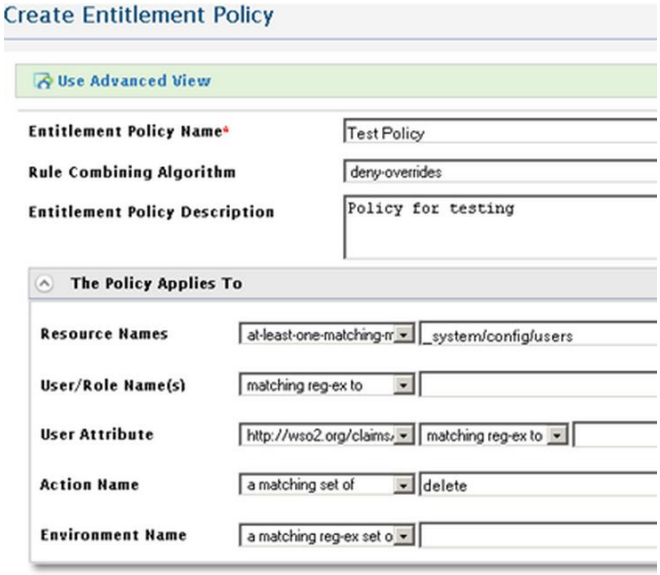

그림 4. 기존의 정책 생성 도구

현존하는 정책 생성 도구는 대부분 웹 기반이며 텍스트, 체크박스, 스크롤 선택자로 이루어져 있다 [5]. 그림 4 는 WSO2 Identity Server 에서 정책을 생성하기 위해 사용되는 Policy Editor 의 외관이다. 각 메뉴마다 XACML 구조에 관한 내용을 직접적으로 기입해야 하고 하나의 정책을 생성하는데 10 단계 이상의 과정을 거쳐야 하는 번거로움이 있다. 무엇보다도 기존에 존재하는 정책과 비교하여 새로운 정책을 생성해야 중복 및 충돌을 피할 수 있지만 기존의 정책을 Editor 상에서 바로 볼 수 있는 방법은 없다. 대부분의 정책 생성 도구는 이와 같은 인터페이스로 이루어져 있고 직관적으로 보기 불편할뿐더러 복잡한 정책을 생성할 경우 정책을 정확하게 표현할 수가 없다. 따라서, 이런 도구를 사용하지 않고 정책을 생성하는 것은 앞서 언급한 것과 같이 직접적으로 XACML 을 작성해야 되기 때문에 더욱 어렵고 복잡하다. 무엇보다도 정책이 생성되어도 기존의 정책과 비교, 분석이 어려워 중복 생성 가능성이 존재하고 이는 자원의 낭비와 정책 충돌로 이어질 수 있다. 따라서, 직관적인 정책의 비교와 분석을 가능하게 하는 도식화된 인터페이스가 필요하다.

본 논문에서는 정책 생성 도구를 만들기 위한 첫 번째 단계로 정책 생성시 기존의 정책과 생성하는 정책간의 비교, 분석을 용이하게 해주는 GUI(Graphical User Interface)기반의 직관적이고 쉬운 인터페이스를 가진 XACML 정책 뷰어를 만들고자 한다.

4. GUI 기반의 직관적이고 쉬운 인터페이스

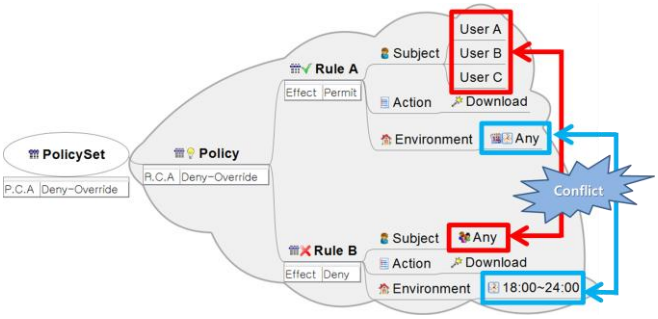

그림 5. GUI 기반의 정책 뷰어로 표현한 ABAC 예제

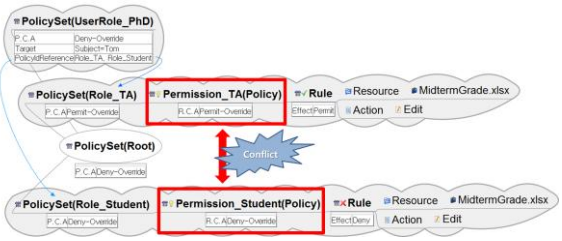

그림 6. GUI 기반의 정책 뷰어로 표현한 RBAC 예제

그림 3 과 같이 복잡한 코드 및 정책을 한눈에 직관적으로 이해하는 것은 어렵다. 하지만, 이를 GUI 기반의 도구로 도식화한다면 한눈에 알아보기 쉬울 뿐만 아니라 코드 및 정책간 비교, 분석도 용이해진다. 따라서 현재 사용되고 있는 다양한 GUI 기반의 도구 중, 누구나 사용하기 쉽고 확장 가능한 마인드맵을 이용하여 XACML 로 작성된 정책을 도식화할 수 있는 정책 뷰어를 설계하였다.

그림 5, 6 은 마인드맵을 확장하여 만든 정책 도구의 초기 모델인 정책 뷰어로써 이전에 제시한 예시를 도식으로 나타내었다. PolicySet, Policy, Rule 은 필요한 알고리즘, Target 등의 속성을 가지고 있고 각각의 Rule 은 Effect 속성과 함께 주체, 행동, 환경의 요소를 가지고 있다. 이와 같이 도식화하여 나타냄으로써 한눈에 정책의 비교, 분석할 수 있게 해준다. 특히 도식화를 함으로써 위의 빨간 사각형 부분의

정책이 충돌한다는 사실을 발견할 수 있다. 이와 같이 코드로 XACML 정책을 보는 것보다 정책을 도식화하면 더욱 직관적으로 볼 수 있다.

위와 같이 GUI 기반의 정책 뷰어를 사용함으로써 기존에 존재하는 텍스트 기반의 정책 생성 도구에 존재하는 불분명하고 비교, 분석이 어려운 점을 해소할 수 있다. 더 나아가 새로운 정책을 생성할 때, 기존의 정책과 손쉽게 비교, 분석이 되어 충돌을 사전에 예방 할 수 있는 효과까지 얻을 수 있다. 하지만, 복잡하고 계층구조를 가진 정책을 표현하기에는 마인드맵 기반으로는 한계가 있다. 그러한 복잡한 정책을 표현하기 위해서 모듈화가 필요하고 더욱 향상된 표현법이 추가되어야 한다.

5. 향후연구

현재 GUI 기반의 XACML 정책 뷰어의 프로토타입 개발이 완료되었고 대부분의 간단한 정책 생성에 필요한 요소들은 표현이 가능하다. 하지만 대규모 분산 시스템에 사용되는 복잡하고 계층구조를 가진 정책을 표현하기에는 아직 부족하다. 이에 향후 연구로는 접근제어 정책을 더욱 세밀하고 명확하게 표현하기 위한 표현법을 추가하고 복잡한 정책은 카테고리별로 나누어 모듈 단위로 관리할 수 있는 방법론의 개발을 계획하고 있다. 또한 개발된 뷰어를 실제로 XACML 이 사용되는 사례연구에 적용하여 실용성, 호환성, 그리고 편의성 등의 평가를 하고자 한다.

본 연구의 최종적인 목표는 기존의 불편한 사용자 인터페이스를 개선한 하이브리드 사용자 인터페이스를 적용시킨 뷰어와 정책을 생성하는 Editor 를 추가하여 XACML 의 코어 엔진에 접목시켜 보다 쉬운 정책 생성 및 정책간의 충돌을 사전에 예방하는 것이다. 더 나아가 효율성 있는 정책 충돌 알고리즘이 개발된다면 이를 이용하여 도식화 진행과정에서 자동으로 충돌을 탐지하고 경고해주는 기능을 구현할 예정이다.

6. 결 론

최근 스마트 기기의 보급으로 인터넷의 접근도가 높아지고 정보, 즉 자원의 흐름이 가속화되고 있다. 이러한 자원을 효율적으로 관리하기 위하여 대규모 분산 시스템(클라우드 컴퓨팅 환경, 빅데이터를 위한 Hadoop, 구글의 MapReduce 등의 기술)이 도입되어지고 있고 이러한 시스템의 자원에 접근하기 위해서는 세밀한 접근제어가 필수적이다. 세밀한 접근제어를 위한 접근제어 정책 언어로 XACML 이 표준으로 채택되고 현재는 대부분의 시스템에서 활발하게 사용되고 있다.

XACML 은 정책의 생성이 자유로운 반면, 배경지식이 풍부하지 않은 상태에서 부주의하게 정책을 생성하게 되면 기존의 정책과 충돌이 발생하는 문제점이 있다. 이와 같은 문제점은 자원의 노출로 이어지고 심각한 피해를 발생시킬 수 있다. 현재 존재하는 정책 생성 도구는 사용이 어려울뿐더러 기존의 정책과 비교, 분석하기가 어려워 충돌여부를 판단할 수 없다. 따라서, 정책관리자가 정책을 생성할 때 직관적이고 쉽게 정책을 볼 수 있고 동시에 비교, 분석이 가능한 GUI 기반의 정책 관련 도구가 필요하다. 이를 위해 XACML 정책 뷰어가 개발됨으로써 정책을 보다 직관적으로 볼 수 있고 기존의 정책과 생성되는 정책간의 충돌을 사전에 미리 발견할 수 있는 효과를 보였다.

더 나아가 뷰어에 복잡하고 계층화된 정책을 표현하기 위한 표현법이 추가된 하이브리드 사용자 인터페이스가 도입되어야 하고 도식화하는 과정에서 자동으로 정책을 발견하고 경고해주는 기능도 필요할 것이다. 그와 함께 정책 생성 기능을 접목하여 기존에 없는 기능을 가진 새로운 XACML 정책 생성 도구가 만들어져야 할 것이다.

참고문헌

[1] Bo Lang, Nan Zhao, Kun Ge and Kai Chen, “An XACML Policy Generating Method Based on Policy View”, In Proceedings of the 3rd International Conference on Pervasive Computing and Applications, pp.295-301, 2008

[2] Abd EL-Aziz Ahmed Abd EL-Aziz and A, Kannan, “Access Control for Healthcare Data using Extended XACML-SRBAC Model”, In Proceedings of the 3rd International Conference on Computer Communication and Informatics, pp.1-4, 2012

[3] Erik Rissanen(2010, August, 10). eXtensible Access Control Markup Language (XACML) Version 3.0. (1st edition). [On-line]. Available: 73 http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cs-01-en.pdf [December 13, 2013].

[4] Jaejin Kim and Scott Uk-Jin Lee, Conflict Detection for XACML Policies", In Proceedings of the 40th Korea Computer Congress, pp.550-552, 2013

[5] Sanchez, M., Lopez, G., Gomez-Skarmeta, A. F., Canovas, O., "Using Microsoft Office InfoPath to Generate XACML Policies.", In Proceedings of the 3rd International Conferece, ICETE 2006, pp.134-145, 2006.