西安科技大学

**《专业文献翻译与写作》**

**课程报告**

**学院：** 计算机科学与技术学院

**专业：**  网络工程

**班级：** 网工1801

**学号：** 18408020129

**姓名：** 吴斌

**日期：** 2021年7月16日

Title: SBA-XACML: Set-based approach providing efficient policy decision process for accessing Web services

Keywords: Web services，Security，Set-based algebra，Policy evaluation，Real-time decision，Access control，XACML Abstract:Policy-based computing is taking an increasing role in providing real-time decisions and governing the systematic interaction among distributed Web services. XACML (eXtensible Access Control Markup Language) has been known as the de facto standard widely used by many vendors for specifying access and context-aware policies. Accordingly, the size and complexity of XACML policies are significantly growing to cope with the evolution of web-based applications and services. This growth raised many concerns related to the efficiency of real-time decision process (i.e. policy evaluation) and the correctness of complex policies. This paper is addressing these concerns through the elaboration of SBA-XACML, a novel Set-Based Algebra (i.e. SBA) scheme that provides efficient evaluation of XACML policies. Our approach constitutes of elaborating (1) a set-based language that covers all the XACML components and establish an intermediate layer to which policies are automatically converted, and (2) a semantics-based policy evaluation that provides better performance compared to the industrial standard Sun Policy Decision Point (PDP) and its corresponding ameliorations. Experiments have been conducted on real-life and synthetic XACML policies in order to demonstrate the efficiency, relevance and scalability of our proposition. The experimental results explore that SBA-XACML evaluation of large and small sizes policies offers better performance than the current approaches, by a factor ranging between 2.4 and 15 times faster depending on policy size.

Introduction:Web and cloud services are becoming very popular and constituting the primary techniques for data exchange between distributed systems and partners. Nowadays, several services are being composed (Karakoc & Senkul, 2009; Mourad, Ayoubi, Yahyaoui, & Otrok, 2012) or grouped together into communities (Khosrowshahi Asl, Bentahar, Mizouni, Khosravifar, & Otrok, 2014) in order to form complex systems and provide advanced set of features over the web. However, researchers are still facing the risk of exploits due to the vast accessibility of these services over the Internet (Bhalla & Kazerooni, 2007; Wang, Wang, Xu, Kit Wan, & Vogel, 2004). Moreover, critical services are emerging such as banking and other business transactions, which raise many security challenges. In this regard, policy-based computing (Ayoubi, Mourad, Otrok, & Shahin, 2013; Tout, Mourad, & Otrok, 2013; Yahyaoui, Mourad, AlMulla, Yao, & Sheng, 2012) is taking an increasing role in governing the systematic interaction and composition among distributed services. Particularly, access control is the most challenging aspect of Web service security to determine which partner can access which service. Currently, an increasing trend is to declare policies in a standardized specification language such as XACML, the OASIS standard eXtensible Access Control Markup Language (Moses, 2011). XACML has been known as the de facto standard widely used by many vendors for specifying access control and context-aware policies. It has been emerged as alternative solution to the traditional way of embedding policy verification as part of the application features. XACML is an XML-based standard for communicating and enforcing access control policies between services and servers (Ayoubi et al., 2013; Moses, 2011). The XACML based policy has complex structure partitioned into three layers: The top layer contains policy sets, the middle layer contains policies and the lower layer contains rules. Each of the three layers has its own target, which contains a set of subjects, resources and actions. Every policy set has a combining algorithm to make the final decision in case of a tie between its policies, and every policy has a combining algorithm to make the final decision in case of a tie between its rules. According to the current XACML engine (Moses, 2011), each request is submitted to the Policy Enforcement Point (PEP) that formulates it using XACML language. Consequently, the Policy Decision Point (PDP) checks at runtime the request with respect to the policy in order to determine access or deny decision. The final decision is enforced by the PEP. This whole process is referred to by policy evaluation. Please refer to Section 7 Listings 1 and 2 for a complete example of a Bank service XACML policy and request evaluation. With the increase of adopting single, composed and grouped Web services into web-based solutions (Karakoc & Senkul, 2009; Khosrowshahi Asl et al., 2014; Mourad et al., 2012), the size and complexity of XACML policies are significantly growing to cope with this evolution and cover the variety of access conditions. Some reallife composed and grouped policies may nowadays embed hundreds and even thousands of rules. On the other hand, this growth raises many concerns related to the efficiency of real-time decision process of complex policies and makes them candidate for insertion of possible flaws between policies and rules. To elaborate more, XACML evaluation engine is responsible of verifying all the rules of all the participating policies, in addition to resolving their corresponding combining algorithms, in order to handle the decisions to the requests at runtime. Hence, enforcing large size XACML policies will decrease the efficiency of policy evaluation engine, and consequently may create performance bottleneck for the services. Several approaches (Liu, Chen, Hwang, & Xie, 2008; Marouf, Shehab, Squicciarini, & Sundareswaran, 2011; Ngo, Makkes, Demchenko, & de Laat, 2013; Pina Ros, Lischka, & Gómez Mármol, 2012) have been proposed to ameliorate the performance of policy evaluation process of the original XACML engine (Moses, 2011). However, these propositions entail major modification on the Sun PDP architecture (Moses, 2011) and assumptions in terms of continuous policy loading and cumulative reception of all requests, which do not always hold in real world environment and limit their efficiency and usefulness. More details about these limitations are presented in Section 2. Hence, decreasing the overhead of XACML evaluation process still constitutes a real challenge. In this paper, we address the aforementioned accuracy and performance problems by elaborating a novel set-based approach for the evaluation of XACML policies. The formal specification of policies and rules using sets is allowing us to efficiently perform evaluation and analysis tasks. The proposed SBA-XACML scheme is composed of a formal algebra language including an automatic converter and compiler, and a policy evaluation module based on formal semantics. All the approach components have been implemented in one development framework that accepts XACML policies and requests as inputs, converts them automatically to SBAXACML constructs when needed and evaluates the requests and policies to provide the final access decision. To download and get additional information about the developed framework and experiments, please visit the following link: http://www. azzammourad.org/#projects. In this context, the main contributions of SBA-XACML are three folds: Set-based intermediate representation of XACML constructs into readable mathematical syntax that maintains the same XACML policy structure and accounts for all its elements and their sub elements including rule conditions, obligations, request and response. The corresponding language and compiler offer automatic and optional conversion from XACML to SBA-XACML constructs. Formal semantics and algorithms that take advantage of the mathematical operations to provide efficient policy evaluation. Unlike current literature, the adopted approach maintains the same architecture of the industrial standard XACML Sun PDP (Moses, 2011) and respects the major properties and assumptions of real-life environments in terms of remote policy loading upon need and disjoint reception of requests from distributed parties. The experimental results conducted on real-life and synthetic XACML policies explore that SBA-XACML evaluation of large and small size policies provide better performance than Sun PDP (Moses, 2011) and its corresponding ameliorations in the literature (Liu et al., 2008; Marouf et al., 2011; Ngo et al., 2013; Pina Ros et al., 2012), by a factor ranging between 2.4 and 15 times faster depending on policy size. The rest of the paper is organized as follows. The related work is summarized in Section 2. Section 3 is devoted for the approach overview and architecture. The description of the proposed SBAXACML language is presented in Section 4. The formal semantics of SBA-XACML policy evaluation is offered in Section 5. In Section 6, the evaluation algorithms are presented. A case study of policy evaluation and a discussion of the experimental results are illustrated in Sections 7 and 8. Finally, the conclusion is presented in Section 9.

题目: SBA-XACML:基于集的方法，为访问Web服务提供了有效的策略决策过程

关键词：Web服务、安全、基于集合代数、政策评估、实时决策、访问控制、XACML

摘要：基于策略的计算在提供实时决策和管理分布式Web服务之间的系统交互方面发挥着越来越重要的作用。XACML(可扩展访问控制标记语言)已经成为许多供应商广泛使用的事实标准，用于指定访问和上下文感知策略。因此，XACML策略的规模和复杂性正在显著增长，以应对基于web的应用程序和服务的发展。这种增长引起了许多与实时决策过程(即政策评估)的效率和复杂政策的正确性有关的关注。本文通过详细阐述SBA-XACML来解决这些问题，SBA-XACML是一种新的基于集的代数(即SBA)方案，它提供了XACML策略的有效评估。我们的方法构成的详细说明(1)一种基于集合的语言,涵盖所有的XACML组件,建立一个中间层政策自动转换.和(2一个基于语义的政策评估,提供更好的性能相比,工业标准的太阳策略决策点(PDP)及其相应的经验。为了证明我们的命题的效率、相关性和可伸缩性，我们在现实生活和综合XACML策略上进行了实验。实验结果表明，SBA-XACML对大小策略的评估提供了比当前方法更好的性能，根据策略大小的不同，其速度要快2.4到15倍。

引言：Web和云服务正在变得非常流行，并构成了分布式系统和合作伙伴之间数据交换的主要技术。现在，为了形成复杂的系统并在web上提供高级的功能集，一些服务被组合或分组到社区中。然而，由于这些服务在Internet上的广泛可访问性，研究人员仍然面临着被利用的风险。此外，关键服务正在出现，如银行和其他商业交易，这带来了许多安全挑战。在这方面，基于策略的计算在管理分布式服务之间的系统交互和组合方面起着越来越重要的作用。特别是，访问控制控制是Web服务安全中最具挑战性的方面，以阻止挖掘哪个合作伙伴可以访问哪个服务。目前，越来越多的趋势是使用标准化规范语言(如OASIS标准的可扩展访问控制标记语言XACML)来声明策略。XACML是许多供应商广泛使用的事实标准，用于指定访问控制和上下文感知策略。它已经出现作为替代解决方案的传统方式嵌入叮策略验证作为应用程序功能的一部分。

XACML是一个基于xml的标准，用于在服务和服务器之间通信和执行访问控制策略。基于XACML的策略具有复杂的结构，分为三层:顶层包含策略集，中间层包含策略，下层包含规则。这三层中的每一层都有自己的目标，目标包含一组主题、资源和操作。每个策略集都有一个组合算法，在其策略之间的匹配情况下做出最终决策，并且每个策略都有一个组合ing算法，在其规则之间的匹配情况下做出最终决策。根据当前的XACML引擎，每个请求都被提交给策略实施点(Policy Enforcement Point, PEP)， PEP使用XACML语言来制定请求。因此，策略决策点(PDP)在运行时检查与策略相关的请求，以确定访问或拒绝决策。最终决定由PEP执行。这整个过程被政策评估所引用。有关Bank服务XACML策略和请求评估的完整示例，请参阅第7节清单1和2.

随着采用单一、组合和分组的Web服务到基于Web的解决方案的增加(Karakoc & Senkul, 2009; Khosrowshali Asl等人，2014;Mourad等人，2012)，XACML策略的规模和复杂性正在显著增长，以应对这种演变，并覆盖各种访问条件。现在，一些实际的组合和分组策略可能嵌入数百甚至数千条规则。另一方面，这种增长引起了对复杂策略实时决策过程效率的关注，并使其成为策略和规则之间可能存在缺陷的候选。更详细地说，XACML评估引擎负责验证所有参与策略的所有规则，并解析它们对应的组合算法，以便在运行时处理请求的决策。因此，实施大型XACML策略将降低策略评估引擎的效率，从而可能为服务创建性能瓶颈。几种方法(刘、陈、黄、谢，2008;MaroufShehab，Squicciarimi，& Sundareswaran,2011;非政府组织，Makes,Demchenko和de Laat,2013;Pina Ros,Lischka，& Gomez Marmol 2012)提出了改进原XACML引擎政策评估过程的性能(Moses，2011)。然而，这些主张需要对Sun PDP体系结构进行重大修改(Moses，2011)，并在所有请求的持续策略加载和累积接收方面做出假设，这些在现实环境中并不总是存在，并限制了它们的效率和有用性。关于这些限制的更多细节将在第2节中介绍。因此，减少XACML评估过程的开销仍然是一个真正的挑战。

在本文中，我们通过阐述一种新的基于集的方法来评估XACML策略，从而解决了上述的准确性和性能问题。使用集合的策略和规则的正式规范允许我们有效地执行评估和分析任务。提出的SBA-XACML方案由包含自动转换器和编译器的形式化代数语言和基于形式化语义的策略评估模块组成。所有方法组件都在一个开发框架中实现，该框架接受XACML策略和请求作为输入，在需要时自动将它们转换为SBA-XACML构造，并评估请求和策略以提供最终的访问决策。要下载和获取有关开发的框架和实验的其他信息，请访问以下链接.http/www。 azzammourad.orgl #项目。在这种情况下，SBA-XACML的主要贡献有三方面:

将XACML构造的基于集的中间表示转换为可读的数学语法，该语法维护相同的XACML策略结构，并解释其所有元素及其子元素(包括规则条件、义务、请求和响应)。相应的语言和编译器提供了从XACNL到SBA-XACML构造的自动可选转换。

利用数学运算提供有效策略评估的形式语义和算法。与当前的文献不同，所采用的方法维护了工业标准XACML Sun PDP的相同体系结构，并考虑了现实环境的主要属性，并在需要时进行远程策略加载和不连接地接收来自分布式方的请求方面假设了。在现实生活和综合XACML策略上进行的实验结果表明，SBA-XACML评估大小和大小策略提供了比Sun PDP更好的性能，根据策略大小的不同，SBA-XACML相应的改进速度是Sun PDP的2.4到15倍.

本文的其余部分组织如下。第2节对相关工作进行了总结。第3节专门介绍方法概述和体系结构。建议的SBAXACML语言的描述在第4节中给出。SBA-XACML策略评估的正式语义将在第5节中提供。在第6节中，给出了评估算法。第7节和第8节阐述了政策评价的案例研究和对实验结果的讨论。最后，在第9节给出结论.

写作：

The core content of computer network learning is the learning of network protocol. A network protocol is a set of rules, standards, or conventions established for data exchange in a computer network. Because the data terminals of different users may adopt different character sets, the two need to communicate, which must be carried out on a certain standard. A vivid metaphor is our language. We have a large territory, a large population and a rich range of local languages, with a huge gap between dialects. The dialects in area A may not be acceptable to people in area B, so we need to establish A language standard for people to communicate with each other throughout the country, and this is what our Mandarin is for. Similarly, looking at the world, the standard language for us to communicate with foreign friends is English, so we have to reluctantly learn English.

Computer network protocols are as diverse as our languages. ARPA's 1977-1979 launch of a network protocol called ARPANET was widely welcomed, most notably because it introduced the well-known TCP/IP standard network protocol. At present, TCP/IP protocol has become the "common language" in the Internet.

In order to make the computers produced by different computer manufacturers communicate with each other, so as to establish a computer network in a larger scope, the International Organization for Standardization (ISO) put forward the "Open System Interconnection Reference Model" in 1978. The famous OSI/RM Model (Open System Interconnection Reference Model).

译文：

计算机网络学习的核心内容就是网络协议的学习。网络协议是为计算机网络中进行数据交换而建立的规则、标准或者说是约定的集合。因为不同用户的数据终端可能采取的字符集是不同的，两者需要进行通信，必须要在一定的标准上进行。一个很形象地比喻就是我们的语言，我们大天朝地广人多，地方性语言也非常丰富，而且方言之间差距巨大。A地区的方言可能B地区的人根本无法接受，所以我们要为全国人名进行沟通建立一个语言标准，这就是我们的普通话的作用。同样，放眼全球，我们与外国友人沟通的标准语言是英语，所以我们才要苦逼的学习英语。

计算机网络协议同我们的语言一样，多种多样。而ARPA公司与1977年到1979年推出了一种名为ARPANET的网络协议受到了广泛的热捧，其中最主要的原因就是它推出了人尽皆知的TCP/IP标准网络协议。目前TCP/IP协议已经成为Internet中的"通用语言"。

为了使不同计算机厂家生产的计算机能够相互通信，以便在更大的范围内建立计算机网络，国际标准化组织（ISO）在1978年提出了"开放系统互联参考模型"，即著名的OSI/RM模型（Open System Interconnection/Reference Model）。