ISSN 2350 - 0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 1, Issue 1, August 2014

Discovery and Resolution of Anomalies in Web Access Control Policies

Banupriya.K, Jenipa.R, Jeyashree.P, Revathi.M

P.S.R.Rengasamy College of Engineering , Anna University, Chennai, India

ABSTRACT: Web services and cloud computing perform business services more efficiently and effectively ,however we still suffer from inadvertent security leakage so in orded to avoid this , an pioneering policy anomaly analysis approach for Web access control policies is focusing on XACML (eXtensible Access Control Markup Language) policy .An policy based segmentation technique to precisely identify policy anomalies and derive effective anomaly resolutions, along with an spontaneous visualization representation of analysis results.An proof-of-concept accomplishment of our method called XAnalyzer and demonstrate how our loom can professionally discover and resolve policy anomalies.

KEYWORDS: Access control policies, XACML, conflict, redundancy, discovery and resolution.

I.INTRODUCTION

The terrific growth of Web applications and Web services set out on the Internet, uses a policy based approach has recently received considerable awareness to hold the security requirements covering large, open, scattered and assorted computing Environments. XACML (eXtensible Access Control Markup Language) [5], which is a universal access control policy language standardized by the Organization for the Advancement of Structured Information Standards (OASIS),has been broadly adopted to specify access control policies for various applications ,particularly Web services .

In an XACML policy, multiple rules may overlap, which means one access request may match several rules. Furthermore, several rules within one policy may conflict, implying that those rules not only overlap each other but also yield different decisions. Conflicts in an XACML policy may lead to both security problem and availability problem.

An intuitive means for resolving policy conflicts by a policy designer is to eliminate all conflicts by modifying the policies. Yet, resolving conflicts through changing the policies is clearly complex, even impractical, in practice from many aspects. Primiarly, the number of conflicts in an XACML policy is potentially large, since an XACML policy may consist of hundreds or thousands of rules. Subsequent, conflicts in XACML policies are most likely very complicated, for the reason that one rule may conflict with multiple other rules, and o ne conflict may be coupled with several rules. Further, an XACML policy for a disseminated application may be aggregated from various parties. In addition, an XACML policy may be maintained by more than one administrator. Without a priori knowledge on the original intentions of policy specification, changing a policy may affect the policy's semantics and may not resolve conflicts correctly. Another critical problem for XACML policy analysis is redundancy discovery and removal. A rule in an XACM policy is unnecessary if every access request that matches the rule also matches other rules with the same effect. Consequently, policy redundancy is treated as policy anomaly as well. Redundancy eradication can be regarded as one of valuable solutions for optimizing XACML policies and educating the appearance of XACML evaluation.

ISSN 2350 - 0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 1, Issue 1, August 2014

II.BACKGROUND

A. Overview of XACML

XACML has turn into the de facto standard for describing access control policies and offers a large set of built-in functions, data types, combining algorithms, and usual profiles for defining application-specific features. At the origin of all XACML policies is a policy or a policy set. A policy set is composed of a progression of policies or other policy sets along with a policy combining algorithm and a target. A policy represents a single access control policy expressed through a target, a set of rules and a rule combining algorithm. The target defines a set of subjects,resources and actions the policy or policy set applies to.A rule set is a sequence of rules. Each rule consists of a target, a condition, and an effect. The target of a rule determines whether an access request is applicable to the rule and it has a similar structure as the target of a policy or a policy set.

An XACML policy often has conflicting rules or policies, which are resolved by four different combining algorithms: Deny-Overrides, Permit-Overrides, First-Applicable and Only-One-Applicable [5].

B. Anomalies in XACML Policies

An XACML policy may contain both policy components and policy set components. While addressing an XACML policy anomalies it involves both policy level and policy set level.

• Anomalies at Policy Level: A rule is conflicting with other rules, if this rule overlaps with others but defines a different effect.

• Anomalies at Policy Set Level: Anomalies may also occur across policies or policy sets in an XACML policy.

III.UNDERLYING DATA STRUCTURE

Our policy-based segmentation procedure introduced in consequent sections requires a well-formed representation of policies for performing a variety of set operations. Binary Decision Diagram (BDD) [3] is a data structure that has been widely used for formal verification and simplification of digital circuits. BDD is the fundamental data structure to symbolize XACML policies and facilitate effective policy analysis.BDDs are acyclic directed graphs which represent Boolean expressions compactly. Each non terminal node in a BDD represents a Boolean variable, and has two edges with binary labels, 0 and 1 for nonexistent and existent, respectively. Terminal nodes represent Boolean value T(True) or F (False).

IV.CONFLICT DETECTION AND RESOLUTION

A. Conflict detection approach

The conflict detection mechanism examines conflicts at both policy level and policy set level for XACML policies.In order to accurately identify policy conflicts and facilitate an effective conflict resolution, a policy-based segmentation technique to partition the entire authorization space of a policy into disjoint authorization space segments. Then, conflicting authorization space segments which contain policy components with different effects, are identified. Each conflicting segment indicates a policy conflict.

B. Conflict detection at policy level

A policy component in an XACML policy includes a set of rules. Each rule defines an authorization space with the effect of either permit or deny. Use an authorization space with the effect of permitted space and an authorization space with the effect of deny denied space. To facilitate the correct interpretation of analysis results, a crisp and sensitive representation

ISSN 2350 - 0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 1, Issue 1, August 2014

method is essential. For the purposes of briefness and understandability, first an XACML policy typically has multiple fields, thus a complete representation of authorization space should be multi-dimensional.

B.1. Conflict Detection at Policy Set Level

There are two major components that need to be taken into consideration when a design an approach for XACML analysis at policy set level:

1. XACML have four rule/policy combining algorithms:
First-Applicable,Only-One Applicable, Deny-Overrides, and Permit-Overrides.
2. An XACML policy is specified recursively and,therefore, has a hierarchical structure.

XACML,a policy set contains a sequence of policies or policy sets, which may further contain other policies or policy sets.

1. CA=First-Applicable. In this case, the effect of a
conflicting segment equals to the effect of the firstcomponent covered by the conflicting segment.

2. CA= Permit-Overrides. The effect of a conflicting segment is always assigned with “Permit,” sincethere is at least one component with “Permit” effect within this conflicting segment.

3. CA = Deny-Overrides. The effect of a conflicting segment always equals to “Deny.”

4. CA = Only-One-Applicable. The effect of a conflicting segment equals to the effect of only applicable component.by the owner (a policy or a policy set) of the segment:

C. Fine-Grained Conflict Resolution

Once conflicts within a policy component or policy set component are identified, a policy designer can choose appropriate conflict resolution strategies to resolve those identified conflicts. First, existing conflict resolution mechanisms in XACML are too restrictive and only allow a policy designer to select one combining algorithm to resolve all identified conflicts within a policy or policy set component.A policy designer may want to adopt different combining algorithms to resolve different conflicts. Second, XACML offers four conflict resolution strategies. However, manyconflict resolution strategies exist but cannot be specified in XACML.

VI .REDUNDANCY DISCOVERY AND REMOVAL

Redundancy discovery and removal mechanism also leverage the policy-based segmentation technique to explore redundancies at both policy level and policy set level.

A. Authorization Space Segmentation

It perform the policy segmentation Partition to divide the entire authorization space of a policy into disjoint segments. classify the policy segments in following categories: nonoverlapping segment and overlapping segment, which is further divided into conflicting overlapping segment and nonconflicting overlapping segment. Each nonoverlapping segment associates with one unique rule, and each overlapping segment is related to a set of rules, which may conflict with each other or have the same effect.

ISSN 2350 - 0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 1, Issue 1, August 2014

B. Irremovable Rule Identification Considering Multivalued Requests

An XACML request may be multivalued. For example, an XACML request can be "a person, who is both a Developer and a Designer, wants to change reports," where the subject has two values, Developer and Designer.

C. Property assignment for rule subspaces

Property task for rule subspace is covered by a policy division is assigned with a property.Four property values, Removable(R),Strong Irremovable(SI) ,Weak Irremovable and Correlated(C),are defined to reflect different characteristics of rule subspace.Removable property is used to indicate that a rule subspace is removable.Removable such a rule subspace does not make any crash on the original authorization space of an associated policy.Strong irremovable property means that a rule subspace cannot be removed because subspace belongs to an irremovable rule with respect to multi-valued requests,and the effect of corresponding policy segment can be only decided by this rule.Weak irremovable property is assigned to a rule subspace when any subspace belonging to the same rule has strong irremovable property.A rule subspace becomes irremovable due to the reason that other portions of this rule cannot be removed.Correlated property is assigned to multiple rule subspaces is covered by a policy segment.

D. Rule correlation break and redundancy removal

Rules covered by an overlapping segment are connected with each other when the effect of the overlapping segment can be determined by any of those rules.Thus,keeping one connected rule and removing others do not change the effect of the overlapping segment.Some rules may get involved in multiple correlated relations.The goal of rule connection break is to break as many redundant rules as possible.Different sequences to break rule correlations may lead to different results for redundancy removal and we can break this connected relations into different sequences.

I. Redundancy elimination at policy set level

The solution of dissimilarity detection at policy set level.The redundancy removal for a policy set is based on an XACML tree structure representation.After each component of a policy set (PS) performs redundancy removal function.

The authorization space of PolicySet can be then partitioned into disjoint segments by performing partition() function.In the solution for conflict detection at policy set level,the total agreement subspaces of each of each child node before performing space partition,because only need to identify conflicts among children nodes to guide the selection of policy combining algorithms for the policy set.

VII..IMPLEMENTATION AND EVALUATION

The implementation of a policy analysis tool is called a XAnalyzer in Java.Based on our policy anomaly analysis mechanism, it consists of four core components **:** segmentation module,effect constraint generation module,strategy mapping module,and property assignment module.The segmentation module takes XACML policies as an input and identifies the authorization space segments by partitioning the authorization space into disjoint subspaces.XAnalyzer utilizes.APIs provided by Sun XACML implementation is to parse the XACML policies and construct Boolean encoding. The effect constraint generation module takes conflicting segments as an input and generates effect constraints for each conflicting segment.Effect constraints are generated based on strategies assigned to each conflicting segment. The property assignment module automatically assigns corresponding property to each subspace covered by the segments of XACML policy components.The assigned properties are in turn utilized to identify redundancies. The estimation is to conflict

ISSN 2350 - 0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 1, Issue 1, August 2014

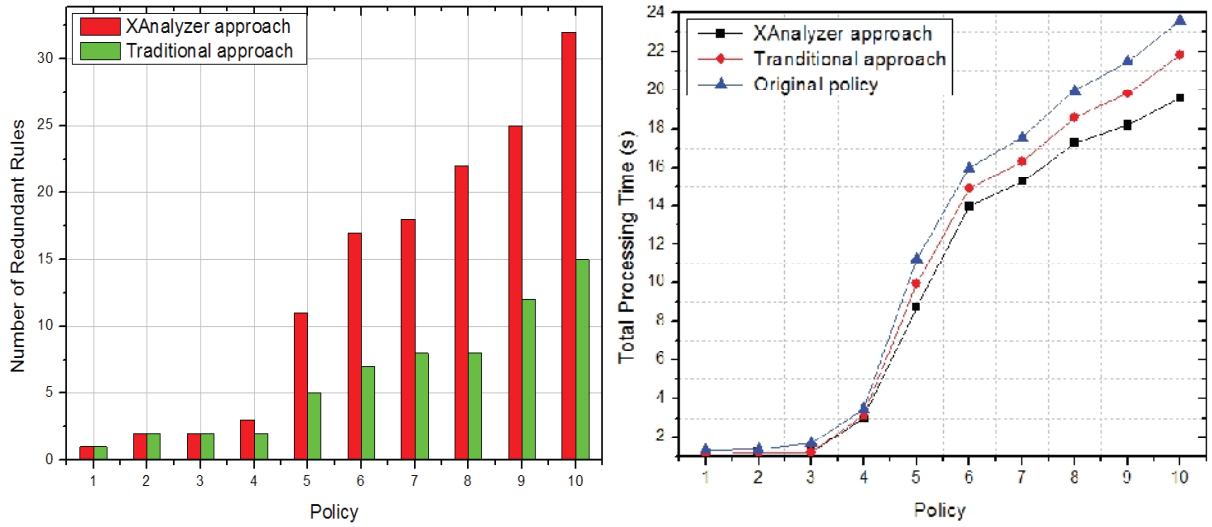
detection approach and the evaluation of redundancy removal approach.The estimation is performed at both policy level and policy set level.

A. Evaluation of conflict detection

The time required by XAnalyzer for conflict detection highly depends upon the number of segments generated for each XACML policy.The increase of the number of segments is proportional to the number of components contained in an XACML policy.It is more efficient in conflict discover approach.

B. Evaluation of redundancy removal

The redundancy removal is considering a single –valued requests and also redundancy removal considering a multi-valued requests.The XACML policies is the redundancy removal approach considering single-valued requests.The multi-valued requests were taken into account in our redundancy removal algorithm and the rules became irremovable.The redundancy analysis algorithm is efficient.The estimation is effective by comparing our redundancy analysis approach with traditional redundancy analysis approach ,which can only identify redundancy relations between two rules.When redundancies in a policy are removed,the performance of policy enforcement is improved.For each of XACML policies ,the total processing time responding is 10,000 generated in XACML requests.The evaluation results clearly show that the processing times are reduced after eliminating redundancies in XACML policies applying our approach can obtain better performance improvement.



VIII.CONCLUSION

These mechanisms can be used to detection and resolution of XACML policyanomalies. A policy-based segmentation mechanism and a grid-based representation technique to effective and efficient anomaly analysis. That a policy designer could easily discover and resolve anomalies in an XACML policy with the help of XAnalyzer. Systematic mechanism and tool will significantly help policy managers support an Web application management service.A future work, the coverage of our approach needs to be further extended with respect to obligations and user defined functions in XACML. To conduct formal analysis [2], [4] of policy anomalies, particularly dealing with multi-valued requests.

ISSN 2350 - 0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 1, Issue 1, August 2014

REFERENCES

[1] D. Agrawal, J. Giles, K. Lee, and J. Lobo. Policy ratification. In *Sixth IEEE International Workshop on Policies for Distributed Systems and Networks, 2005*, pages 223–232, 2005.

[2] G. Ahn, H. Hu, J. Lee, and Y. Meng. Representing and Reasoning about WebAccess Control Policies. In *34th Annual IEEE Computer* Software *and Applications Conference*, pages 137–146. IEEE, 2010.

[3] K. Fisler, S. Krishnamurthi, L. Meyerovich, and M. Tschantz.Verification and change-impact analysis of access-control policies.In *Proceedings of the 27th international conference on Software engineering*, pages 196–205. ACM, 2005.

[4] H. Hu and G. Ahn. Enabling verification and conformance testing for access control model. In *Proceedings of the 13th ACM symposiumon Access control models and technologies*, pages 195–204. ACM,2008.

[5] T. Moses et al. Extensible access control markup language(XACML) version 2.0. *Oasis Standard*, 200502, 2005.