language **en**

LEAR Authorization Vocabulary

Release 30 Juni 2017

This version:
http://www.daspsec.org/o/lear/0.2
Revision:
0.2
Authors:
Bojan Suzic, (Graz University of Technology)
Publisher:
A-SIT Secure Information Technology Center - Austria
Download serialization:
Format RDF/XML Format N Triples Format TTL

Abstract

This document proposes a vocabulary for modeling of security policies using LEAR access control model. It serves as one of the building blocks of DASP framework, whose primary goal is achieving fine-grained cross-domain security management for heterogenous environments and intelligent agents.

Table of contents

1. Introduction

back to ToC

Simplicity, clarity of the structure and its relationship with underlying standards and technologies are some of the factors that allowed RESTful architectural style to become broadly adopted approach in exposing web services. Nowadays, many of the service-based intra and inter-domain interactions rely on Web APIs. Many of them, however, follow RESTful paradigm only in broader sense, combining both RPC-like and resource oriented view on different maturity levels. Based on the adoption and relevance of RESTful style, LEAR access control model has been proposed to enable modeling of security goals and requirements based on unique characteristics and heterogeneous approaches in Web API implementations. This document proposes a vocabulary for LEAR, allowing its integration in cross-domain processes and other semantic frameworks.

1.1. Namespace declarations

Table 1: Namespaces used in the document

ns	<http://creativecommons.org/ns#>
owl	<http://www.w3.org/2002/07/owl#>
dasp-interaction	<http://www.daspsec.org/o/dasp-interaction/>
xsd	<http://www.w3.org/2001/XMLSchema#>
rdfs	<http://www.w3.org/2000/01/rdf-schema#>
o	<http://www.daspsec.org/o/>
dasp-core	<http://www.daspsec.org/o/dasp-core#>
rdf	<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
terms	<http://purl.org/dc/terms/>
vann	<http://purl.org/vocab/vann/>
lear	<http://www.daspsec.org/o/lear#>
dc	<http://purl.org/dc/elements/1.1/>

2. LEAR Authorization Vocabulary: Overview

back to ToC

This ontology has the following classes and properties.

Classes

Complex Effect   Context   Decision Function   Deny   Effect   Indefinite   Permit
Plain Effect   Security Policy   Security Rule

Object Properties

hasContext   hasDecisionFunction   hasEffect   hasSecurityRule   targetsAction
targetsElement   targetsParameter   targetsResource

Data Properties

restricts   valueEquals   valueGreaterThan   valueLowerThan   valueNotEquals

3. LEAR Authorization Vocabulary: Description

back to ToC

Policy languages are applied to specify access control constraints in the system. During the previous decades, many policy-based access control languages have emerged. Such approaches enable externalizing authorization functionality from the applications, supporting the separation of concerns between application developers and security experts while improving the overall maintainability of the system. This is achieved by modularizing the components and separating the functionality so that each component can be adjusted without significantly affecting the whole system. This way, the policies can be dynamically adjusted in a running system to reflect evolving security goals.

One of access control models that attracted significant attention in its several instantiations is Attribute-based Access Control (ABAC). XACML [XACML] as a set of standardized specifications has become the dominant approach in specifying ABAC policies. The primary motivation behind XACML is the consolidation of security enforcement within a single (large) enterprise. Our contribution, in the form of LEAR access control model and authorization vocabulary, focuses on establishing externalized authorization that goes beyond organizational boundaries, achieves flexible deployments and lowers adoption obstacles. This goal is facilitated by considering LEAR from the ground-up as a concept that depends on cross-organizational processes and related semantic interoperability. This vocabulary is hence provided as an integral part of LEAR model, supporting its integration with various systems and allowing unified policy management that goes beyond a single environment.

LEAR vocabulary is a part of a broader initiative implemented under Data Sharing and Processing Framework (DASP). DASP consists of a conceptual framework, and architectural and interaction models. These entities are complemented with semantic vocabularies, supporting tools and validation models that allow implementation and integration of the framework. Hence, LEAR is one of provided vocabularies that seamlessly integrate with DASP and its tooling stack.

4. Cross reference for LEAR Authorization Vocabulary classes, properties and dataproperties

back to ToC

This section provides details for each class and property defined by LEAR Authorization Vocabulary.

4.1. Classes

Complex Effect^c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#ComplexEffect

Encompasses effects that define complex decisions, potentially involving additional functionality or implying the execution of additional steps in order to fully resolve the rule effect.

has super-classes

Effect ^c

Context^c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#Context

Represents a particular situation. Context is typically used to express conditions or facts present in the system. It can refer to *intrinsic* context, which relates to the conditions that describe internal system states, or to *extrinsic* context, which deals with the conditions provided by the factors external to the system.

has super-classes

thing ^c

Decision Function^c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#DecisionFunction

A function that is applied over a set of rule effects in order to resolve conflicts and derive a final effect of the policy

has super-classes

thing ^c

Deny c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#Deny

Refers to the rejective decision for authorization request. The plain category of effect implies that no further requirements or obligations are imposed to explicity refuse the authorization request.

has super-classes

Plain Effect c

Effect c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#Effect

Effect is resolved or triggered in the case of successful rule match

has super-classes

thing c

has sub-classes

Complex Effect c, Plain Effect c

Indefinite c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#Indefinite

Refers to undecided effect, which usually occurs when there is no enough information available to decide an authorization request.

has super-classes

Plain Effect c

Permit c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#Permit

Refers to the permisive decision for authorization request. The plain category of effect implies that no further requirements or obligations are imposed to allow the authorization request.

has super-classes

Plain Effect c

Plain Effect c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#PlainEffect

Encompasses effects that define straightforward decisions.

has super-classes

Effect c

has sub-classes

Deny c, Indefinite c, Permit c

Security Policy^c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#SecurityPolicy

Represents a minimal object of security evaluation expressed as a set of security rules. Security policy is further defined by *Decision Function*, which allows to determine the authorization effect of several *Security Rules* defined under single security policy

has super-classes
thing ^c

Security Rule^c

back to ToC or Class ToC

IRI: http://www.daspsec.org/o/lear#SecurityRule

Represents a particular security goal that exists on intersection between various events, resources and situational conditions.

has super-classes
thing ^c

is in domain of
hasContext ^op, targetsResource ^op

4.2. Object Properties

hasContext^op

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#hasContext

Contextual condition that is used to activate the rule.

has super-properties
top object property

has domain
Security Rule ^c

has range
context ^c

hasDecisionFunction^op

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#hasDecisionFunction

Decision function is applied over a set of rule effects in order to resolve conflicts and derive a final effect of the policy.

has super-properties
top object property

hasEffect^{op}

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#hasEffect

An *effect* is resolved in a case of successfull rule match. The concept of effect particularly encompasses different classes which can be categorized as *plain* and *complex* effects. Plain effects imply straightforward decisions, such as permit, deny and similar. Complex effects refer to authorization decisions that encompass additional functionality. This may include a class of transformational effects, which extend permit class with additional operations that have to be executed in the course of interaction.

has super-properties
top object property

hasSecurityRule^{op}

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#hasSecurityRule

Associates *security rule* with the *security policy* that encompass one or more rules and formulates a particular security goal.

has super-properties
top object property

targetsAction^{op}

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#targetsAction

Associates an *action* with *security rule* or a *context*.

Applied for security rules, this property implies the target action in the system or client's intent that is a subject of a particular security rule.

When associated with the context, this property provides additional information on how to resolve a particular condition. This may involve the reasoning on the input or output data for the action.

has super-properties
top object property

has domain
intrinsic context ^{c} **or** Security Rule ^{c}

has range
action ^{c}

targetsElement^{op}

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#targetsElement

Associates a *contextual condition* with an *element* exposed by the API. This provides an information to the contextual handler on which element is subjected to the contextual evaluation. Used together with *targetsAction*, it allows the reasoning over and extraction of particular element present in the output of or input to the action.

has super-properties
top object property

has domain
intrinsic context ^{c}

has range
element ^{c}

targetsParameter op

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#targetsParameter

Associates a *context* with a parameter. When the parameter is present in request or response, the contextual condition can be evaluated against its restriction. If it holds, the condition is satisfied and the *security rule* may hold.

has super-properties
top object property

has domain
intrinsic context c

has range
parameter c

targetsResource op

back to ToC or Object Property ToC

IRI: http://www.daspsec.org/o/lear#targetsResource

Associates *security rule* with particular *resource* This allows to relate a security with a resource in the system that might be indirectly affected by some action or operation. In this case the rule is evaluated and, depending on contextual conditions, activated.

has super-properties
top object property

has domain
Security Rule c

has range
resource c

4.3. Data Properties

restricts dp

back to ToC or Data Property ToC

IRI: http://www.daspsec.org/o/lear#restricts

Subsumes a range of data properties that provide various types of value restrictions over contextual conditions.

has super-properties
top data property

has sub-properties
valueEquals dp, valueGreaterThan dp, valueLowerThan dp, valueNotEquals dp

valueEquals dp

back to ToC or Data Property ToC

IRI: http://www.daspsec.org/o/lear#valueEquals

Evaluates if a value of given entity equals to the value of data property. One application example considers *contextual condition* which references some element, resource or parameter as a subject of evaluation.

has super-properties
restricts dp

valueGreaterThan^{dp}

back to ToC or Data Property ToC

IRI: http://www.daspsec.org/o/lear#valueGreaterThan

Evaluates if a value of given entity is greater than the value of data property. One application example considers *contextual condition* which references some element, resource or parameter as a subject of evaluation.

has super-properties

restricts ^{dp}

valueLowerThan^{dp}

back to ToC or Data Property ToC

IRI: http://www.daspsec.org/o/lear#valueLowerThan

Evaluates if a value of given entity is lower than the value of data property. One application example considers *contextual condition* which references some element, resource or parameter as a subject of evaluation.

has super-properties

restricts ^{dp}

valueNotEquals^{dp}

back to ToC or Data Property ToC

IRI: http://www.daspsec.org/o/lear#valueNotEquals

Evaluates if a value of given entity is not equal to the value of data property. One application example considers *contextual condition* which references some element, resource or parameter as a subject of evaluation.

has super-properties

restricts ^{dp}

Legend

back to ToC

^{c}: Classes
^{op}: Object Properties
^{dp}: Data Properties
^{ni}: Named Individuals

5. References

back to ToC

[XACML] eXtensible Access Control Markup Language (XACML) Version 3.0. OASIS standard, OASIS (2013).
URL: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html

[CSA] Cloud Security Open API Working Group: Proposed Charter. CSA Working Group (2015)
URL: https://cloudsecurityalliance.org/group/open-api/

[API16] Riding and thriving on the API hype cycle. Maja Vukovic et al. Communications of ACM 59, 3 (2016)
URL: https://cacm.acm.org/magazines/

[REST14] RESTful or RESTless – Current state of todays top Web APIs. Frederik Bülthoff and Maria Maleshkova. In European Semantic Web Conference. Springer (2014)
URL: http://rdcu.be/tM6B