XACML v3.0 Separation of Duties Version 1.0

Committee Specification Draft 01

14 September 2023

This stage:
https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/csd01/xacml-3.0-duties-v1.0-csd01.docx (Authoritative)
https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/csd01/xacml-3.0-duties-v1.0-csd01.html
https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/csd01/xacml-3.0-duties-v1.0-csd01.pdf

Previous stage:
N/A

Latest stage:
https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/xacml-3.0-duties-v1.0.docx (Authoritative)
https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/xacml-3.0-duties-v1.0.html
https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/xacml-3.0-duties-v1.0.pdf

Technical Committee:
OASIS eXtensible Access Control Markup Language (XACML) TC

Chairs:
Hal Lochhart (harold.w.lochhart@gmail.com), Individual
Bill Parducci (bill@parducci.net), Individual

Editor:
Steven Legg (steven.legg@viewds.com), ViewDS Identity Solutions

Related work:
This document is related to:

- *eXtensible Access Control Markup Language (XACML) Version 3.0 Plus Errata 01.* Edited by Erik Rissanen. 12 July 2017. OASIS Standard incorporating Approved Errata. http://docs.oasis-open.org/xacml/3.0/errata01/os/xacml-3.0-core-spec-errata01-os-complete.html. Latest version: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-en.html.

Abstract:
This specification defines a method for supporting separation of duties within XACML policies using obligations and allowing the full generality of attribute-based access control. In particular, duties are not required to be associated with subject roles.

Status:
This document was last revised or approved by the OASIS eXtensible Access Control Markup Language (XACML) TC on the above date. The level of approval is also listed above. Check the "Latest stage" location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml#technical.

TC members should send comments on this specification to the TC's email list. Others should send comments to the TC's public comment list, after subscribing to it by following the instructions at the "Send A Comment" button on the TC's web page at https://www.oasis-open.org/committees/xacml/.

This specification is provided under the RF on Limited Terms Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC's web page (https://www.oasis-open.org/committees/xacml/ipr.php).

Note that any machine-readable content (Computer Language Definitions) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product's prose narrative document(s), the content in the separate plain text file prevails.

Key words:
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

Citation format:
When referencing this document, the following citation format should be used:

[XACML-Duties-v1.0]

XACML v3.0 Separation of Duties Version 1.0. Edited by Steven Legg. 14 September 2023. OASIS Committee Specification Draft 01. https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/csd01/xacml-3.0-duties-v1.0-csd01.html. Latest stage: https://docs.oasis-open.org/xacml/xacml-3.0-duties/v1.0/xacml-3.0-duties-v1.0.html.

Table of Contents

1 Introduction

[All text is normative unless otherwise labeled]

[Informative]

Separation of duties (SoD) is a security principle applied to minimize fraud, misuse of information, conflicts of interest and user errors by requiring that a task can only be completed by the active involvement of two or more people.

Role-based access control (RBAC) is frequently used to implement separation of duties in information technology systems. Permission to perform each part of a protected task is assigned to a different role and users are prevented from holding all the roles needed to perform the complete task, i.e., the roles are mutually exclusive. Separation of duties relations (also called constraints) define which roles are mutually exclusive.

Separation of duties relations can be further subdivided into static and dynamic relations. Static relations are considered when attempting to assign a role to a user. Static relations prevent a user from holding conflicting roles at any point in time, but do not prevent a user holding conflicting roles at different points in time. The roles assigned to users do change and a user could exploit this to bypass separation of duties. With dynamic relations, users are not statically assigned roles but instead activate the roles they want to use within a “session”. The dynamic relations prevent a user from activating conflicting roles within the same session. For this to enforce separation of duties, the “session” has to encompass all the steps in a task, which may happen over days or weeks, so this isn't what one normally considers to be a “session”.

An RBAC approach to enforcing separation of duties isn’t practical for XACML. An SoD profile could be written as an adjunct to the XACML RBAC profile, but it would only work as long as all user permissions were obtained only via roles. This would be a severe restriction that negates many of the advantages of attribute-based access control. One might as well be using RBAC instead. XACML is also essentially stateless and there isn't an inherent concept of a session of any sort, let alone one that persists for weeks.

This profile takes a different approach to separation of duties by recognizing that conflicting roles are only a proxy for conflicting actions. What we are really interested in doing is preventing a user from performing a conflicting action no matter how the permission for that action is obtained. This profile describes a way to do it using obligations **[XACML-v3.0-Errata01-complete]** and XACML entities **[xacml-3.0-nested-ent-v1.0]**. The obligations are used to instruct a PEP to retain information about actions performed on a resource, which is provided in subsequent authorization requests so that XACML policies can deny actions that would be a violation of separation of duties. The retained information is represented as XACML entities.

1.1 Glossary

Action history record

A record of an action performed by a user on a resource, sent by the PDP as an obligation in an authorization response, to be stored by the PEP and sent in subsequent authorization requests for access to the same resource.

Separation of duties constraint (SoD constraint)

A relationship between a set of two or more actions that can be performed on a resource such that no single user is permitted to perform all of those actions (often, no more than one of those actions).

SoD obligations

The obligations defined in this profile that are sent in authorization responses to manage ***action history records***.

Transaction

A series of authorization requests for access to a resource specifying actions that are related by an ***SoD constraint***.

Transaction time limit

The time at which a PEP can assume a ***transaction*** will never be completed so that the ***transaction's action history records*** can be discarded.

1.1.1 Document conventions

The replacement text for the XML entity reference "`&xacml1;`" used in examples is "`urn:oasis:names:tc:xacml:1.0:`".

The replacement text for the XML entity reference "`&xacml2;`" used in examples is "`urn:oasis:names:tc:xacml:2.0:`".

The replacement text for the XML entity reference "`&xacml3;`" used in examples is "`urn:oasis:names:tc:xacml:3.0:`".

2 Separation of Duties Constraints

This profile describes a method for supporting separation of duties (SoD) in XACML **[XACML-v3.0-Errata01-complete]**. Conceptually, an ***SoD constraint*** in XACML is a relationship between a set of two or more actions that can be performed on a resource such that no single user is permitted to perform all of those actions. The relationship is realized through XACML policies rather than some separate construct. A common use case is that no user may perform more than one of the actions, however, this and other possibilities are achieved just through the way the policies are written. Policies are written to deny access if the user has already performed an action that conflicts with the current requested action.

In order for the policies to test a current requested action against previously performed actions, a record of the previous actions is needed. An obligation is defined in Section 7.1 that requires a PEP to save a description of a permitted action performed on a resource and to provide that description in every subsequent authorization request involving that resource. This description is called an ***action history record*** and is represented as an XACML entity. A PEP is chosen to save the history of actions on a resource because it is generally closest to the already persistent resource. The XACML policies are also written to emit the obligation to cause the PEP to record a new successful action.

Another obligation is defined to indicate when a sequence of actions covered by an ***SoD constraint*** has been completed so that the history of actions can be discarded by the PEP. A policy can add time limits to the ***action history records*** to cover the possibility that the sequence is never properly completed.

This profile supports multiple ***SoD constraints*** on the same resource. To this end, ***action history records*** contain a constraint identifier, defined in Section 4.2, that policies can set and examine. Independent sequences of actions covered by the same ***SoD constraint*** applied to the same resource, and overlapping in time, are also supported. Each sequence is referred to here as a ***transaction*** and ***action history records*** include a transaction identifier defined in Section 4.3.

3 Action History Record

The enforcement of ***SoD constraints*** requires a history of actions performed by users (subjects) on a resource to be maintained by the PEP for a non-trivial period of time, potentially days or weeks. Each such action is described by an ***action history record***, which is represented in XACML authorization requests as a value of the `urn:oasis:names:tc:xacml:3.0:data-type:entity` data-type **[xacml-3.0-nested-ent-v1.0]**. A value of the `entity` data-type holds a collection of XACML attributes.

An `entity` value representing an ***action history record*** MUST contain a `urn:oasis:names:tc:xacml:1.0:resource:resource-id` attribute **[XACML-v3.0-Errata01-complete]** to uniquely identify the resource acted upon.

The `entity` value SHOULD contain a `urn:oasis:names:tc:xacml:1.0:action:action-id` attribute **[XACML-v3.0-Errata01-complete]** to indicate the action performed.

The entity value SHOULD contain a `urn:oasis:names:tc:xacml:1.0:subject:subject-id` attribute **[XACML-v3.0-Errata01-complete]** to uniquely identify the end user who requested access.

The `entity` value MUST contain the `constraint-id` attribute defined in Section 4.2.

The `entity` value MUST contain the `transaction-id` attribute defined in Section 4.3.

The `entity` value MAY contain the `time-limit` attribute defined in Section 4.4.

An SoD policy MAY add other attributes to the ***action history record***.

3.1 Action History Record Example

[Informative]

Figure 1 shows an ***action history record*** (a value of the `entity` data-type) in both XML and JSON.

```
XML:

  <AttributeValue DataType="urn:oasis:names:tc:xacml:3.0:data-type:entity">
    <Attribute IncludeInResult="false"
      AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/purchase-order/10147</AttributeValue>
    </Attribute>
    <Attribute IncludeInResult="false"
      AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >raise</AttributeValue>
    </Attribute>
    <Attribute IncludeInResult="false"
      AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id">
      <AttributeValue
        DataType="urn:oasis:names:tc:xacml:1.0:data-type:rfc822Name"
        >alice@example.com</AttributeValue>
    </Attribute>
    <Attribute IncludeInResult="false"
      AttributeId="urn:oasis:names:tc:xacml:3.0:sod:attribute:constraint-id">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >purchase-order-1</AttributeValue>
    </Attribute>
    <Attribute IncludeInResult="false"
      AttributeId="urn:oasis:names:tc:xacml:3.0:sod:attribute:transaction-id">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >f5541707-d098-4d43-a00c-d75c25b8c377</AttributeValue>
    </Attribute>
    <Attribute IncludeInResult="false"
      AttributeId="urn:oasis:names:tc:xacml:3.0:sod:time-limit">
```

```
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#dateTime"
        >2022-10-10T12:00:00Z</AttributeValue>
    </Attribute>
  </AttributeValue>

JSON:

  {
    "Attribute":[{
      "AttributeId":"urn:oasis:names:tc:xacml:1.0:resource:resource-id",
      "DataType":"anyURI",
      "Value":"http://example.com/purchase-order/10147"
    },{
      "AttributeId":"urn:oasis:names:tc:xacml:1.0:action:action-id",
      "DataType":"string",
      "Value":"raise"
    },{
      "AttributeId":"urn:oasis:names:tc:xacml:1.0:subject:subject-id",
      "DataType":"rfc822Name",
      "Value":"alice@example.com"
    },{
      "AttributeId":
        "urn:oasis:names:tc:xacml:3.0:sod:attribute:constraint-id",
      "DataType":"string",
      "Value":"purchase-order-1"
    },{
      "AttributeId":
        "urn:oasis:names:tc:xacml:3.0:sod:attribute:transaction-id",
      "DataType":"string",
      "Value":"f5541707-d098-4d43-a00c-d75c25b8c377"
    },{
      "AttributeId":"urn:oasis:names:tc:xacml:3.0:sod:time-limit",
      "DataType":"dateTime",
      "Value":"2022-10-10T12:00:00Z"
    }]
  }
```

Figure 1 - An action history record in XML and JSON

4 Common Attributes

4.1 The history Attribute

The `urn:oasis:names:tc:xacml:3.0:sod:attribute:history` attribute holds one or more ***action history records*** as values of the `urn:oasis:names:tc:xacml:3.0:data-type:entity` data-type **[xacml-3.0-nested-ent-v1.0]**. The attribute is used in the resource category of an authorization request to convey the current ***action history records*** for the resource.

4.2 The constraint-id Attribute

The `urn:oasis:names:tc:xacml:3.0:sod:attribute:constraint-id` attribute uniquely identifies an ***SoD constraint*** and serves to associate ***action history records*** with the policies that generate and examine them. Policy writers are free to choose any data-type that has a defined `type-equal` function **[XACML-v3.0-Errata01-complete]**. An ***action history record*** SHALL NOT have more than one `constraint-id` value regardless of the data-type.

Policies that generate ***action history records*** through obligations would typically assign the relevant value to the `constraint-id` attribute in an attribute assignment expression of an obligation expression.

Policies for the ***SoD constraint*** would typically select values of the `history` attribute that have the relevant value for the nested `constraint-id` attribute.

In many cases a resource will be subject to only one possible ***SoD constraint*** and the `resource-id` attribute would suffice to identify the related ***action history records***. However, a `constraint-id` attribute is still required in ***action history records*** to simplify the ***SoD obligation*** processing.

Two `constraint-id` attribute values are considered the same if they have the same data-type and are considered equal according to the `type-equal` function for that data-type.

4.3 The transaction-id Attribute

This profile supports users initiating overlapping ***transactions*** on the same resource. The `urn:oasis:names:tc:xacml:3.0:sod:attribute:transaction-id` attribute is provided to distinguish between independent, contemporaneous ***transactions***. The `transaction-id` attribute in an ***action history record*** SHALL have exactly one attribute value. Policy writers are free to choose any data-type that has a defined `type-equal` function, although note that the `get-string-identifier` function is defined (see Section 5.1) to enable policies to generate globally unique ***transaction*** identifiers for the `http://www.w3.org/2001/XMLSchema#string` data-type if required.

In many cases the resource is created to represent the ***transaction*** (e.g., a purchase order), in which case a separate `transaction-id` would not be needed. However, a `transaction-id` attribute is still required in ***action history records*** to simplify the ***SoD obligation*** processing. In such cases, the policy can set the `transaction-id` to a constant value, or to the value of the `resource-id`, rather than generating a unique identifier.

Two `transaction-id` attribute values are considered the same if they are considered equal according to the `string-equal` function **[XACML-v3.0-Errata01-complete]**.

4.4 The time-limit Attribute

The `urn:oasis:names:tc:xacml:3.0:sod:attribute:time-limit` attribute specifies a date and time that is used to calculate the ***transaction time limit*** for a ***transaction***. The `time-limit` attribute in an ***action history record***, if present, SHALL have exactly one attribute value and the data-type of that value SHALL be `http://www.w3.org/2001/XMLSchema#dateTime`.

5 Functions

5.1 The get-string-identifier Function

The `urn:oasis:names:tc:xacml:3.0:function:get-string-identifier` function SHALL take no arguments and SHALL return a globally unique identifier as an `http://www.w3.org/2001/XMLSchema#string` value.

For the purposes of this profile the generated unique identifier only needs to be unique within the scope of a resource, but to support uses outside this profile, and because it is relatively easy to achieve effective global uniqueness with something like a UUID, global uniqueness is specified for the `get-string-identifier` function.

6 Transaction Time Limit

Each subset of the collection of ***action history records*** held by the PEP for a resource with the same `constraint-id` and `transaction-id` attribute values belong to the same ***transaction***. The ***transaction time limit*** for the ***transaction*** is the greatest attribute value from the `time-limit` attributes of its ***action history records***, as determined by the `dateTime-greater-than` function **[XACML-v3.0-Errata01-complete]**. Note that the `time-limit` attribute is optional. If none of the ***transaction's action history records*** has a `time-limit` attribute then the ***transaction*** does not have a time limit.

A PEP is allowed to discard the ***action history records*** for a transaction when the current time exceeds the ***transaction time limit***. The ***action history records*** for a ***transaction*** with no time limit can only be discarded by an explicit `end-history` obligation or by removing the associated resource.

A policy writer can set an absolute time limit for all the actions in a ***transaction*** to be completed by setting the `time-limit` attribute in the first ***action history record*** for the ***transaction*** and omitting the `time-limit` attribute from all subsequent ***action history records***, or the policy writer can keep pushing the ***transaction time limit*** forward with each successful action by providing an updated `time-limit` value in each new ***action history record***.

7 Obligations

Policies use obligations to manage the ***action history records*** held by PEPs.

7.1 The add-history Obligation

The `urn:oasis:names:tc:xacml:3.0:sod:obligation:add-history` obligation specifies an ***action history record*** to be added to a collection of ***action history records*** maintained for the resource identified by the `resource-id` attribute in the ***action history record***. If such a collection does not already exist then one is created and initialized with the processing of this obligation. To satisfy the obligation the PEP MUST additionally be prepared to add the ***action history record*** to any subsequent authorization request for access to the nominated resource until such time as the relevant ***transaction time limit*** is reached, a corresponding `end-history` obligation is received, or the resource is removed. The ***action history record*** is considered active until that time and inactive from that time onwards.

Each attribute assignment of the `add-history` obligation describes one attribute of the ***action history record*** and MUST NOT have a `Category` or `Issuer` XML attribute. Since the `resource-id`, `constraint-id` and `transaction-id` attributes are required for an ***action history record***, the `add-history` obligation MUST have attribute assignments for at least these attributes.

It is an error if the ***action history record*** contains multiple values of the `resource-id` attribute that identify different resources. The PEP MUST treat the obligation as unsatisfiable in this case, make no change to the stored ***action history records*** and deny access to the resource. It is not an error if an ***action history record*** contains multiple values of the `resource-id` attribute and those values identify the same resource.

The PEP MUST treat the obligation as unsatisfiable if the ***action history record*** does not satisfy the requirements in Section 3.

Before a PEP sends an authorization request it MUST add any active ***action history records*** it holds for a resource in the request as values of the `history` attribute in the resource category. In the case of a request for multiple decisions the PEP must perform this addition for each resource category.

Inactive ***action history records*** can be discarded by the PEP at any time. Whether the PEP discards inactive ***action history records*** immediately, or cleans them out periodically or opportunistically (e.g., when updating the resource) is at the discretion of the implementer.

If a resource is short-lived then the ***action history records*** relating to it will also be short-lived. They will become inactive when the resource is deleted, assuming the PEP can reliably detect that deletion. For resources that are long-lived it is desirable to avoid the PEP holding on to ***action history records*** indefinitely because a transaction is never properly completed. The use of the `time-limit` attribute in the `end-history` obligation is RECOMMENDED to guarantee that ***action history records*** don't inexorably accumulate in the PEP.

7.2 The end-history Obligation

The `urn:oasis:names:tc:xacml:3.0:sod:obligation:end-history` obligation specifies that the ***action history records*** for a nominated ***transaction*** are no longer required and the PEP MUST NOT provide them in any subsequent authorization request. The PEP is free to discard the ***action history records*** of the nominated ***transaction***.

The `end-history` obligation MUST have an attribute assignment for each of the `resource-id`, `constraint-id` and `transaction-id` attributes. The obligation applies to the ***action history records*** with the same values for `resource-id`, `constraint-id` and `transaction-id`.

No other attribute assignments are permitted.

The `end-history` obligation is typically emitted at the end of a ***transaction***, e.g., because the final action in the sequence of actions making up the ***transaction*** has been permitted, because a user has requested an action to explicitly abandon or cancel the ***transaction***, or because the ***transaction*** has

been terminated for policy reasons. The obligation lets the PEP know it no longer needs to hold on to the related ***action history records***.

8 Examples

[Informative]

8.1 Purchase Order Example

This example shows a policy for enforcing an SoD constraint that a purchase order must be approved by someone other than the person who raises it.

A purchase order is represented as a resource with the following attributes:

- `urn:oasis:names:tc:xacml:1.0:resource:resource-id`

  A unique URI identifying the purchase order.

- `urn:oasis:names:tc:xacml:3.0:sod:attribute:history`

  The current collection of ***action history records*** for the purchase order, possibly empty.

- Other attributes detailing the purchase order but not referenced by the policy.

An employee is represented as a subject with the following attributes:

- `urn:oasis:names:tc:xacml:1.0:subject:subject-id`

  A unique employee identifier in the form of an email address.

- `urn:example:xacml:department`

  The department to which the employee is assigned.

- `urn:example:xacml:job-title`

  The title of the job performed by the employee.

- Other attributes detailing the employee but not referenced by the policy.

A purchase order effectively represents a single transaction in its own right and there is no sense in which multiple transactions can apply simultaneously to the same purchase order. Consequently, the `transaction-id` attribute is largely uninteresting in this example and is set to the value of the `resource-id` where required.

The one and only SoD constraint in this example is identified with the string constant `purchase-order`.

```
<Policy xmlns="&xacml3;core:schema:wd-17"
  PolicyId="http://example.com/SoD/purchase-orders" Version="1.0"
  RuleCombiningAlgId="&xacml3;rule-combining-algorithm:deny-overrides">

  <Description>
    Policy for SoD constraints applicable to purchase orders.
  </Description>

  <!-- The target restricts applicability to purchase order resources. -->
  <Target>
    <AnyOf>
      <AllOf>
        <Match
          MatchId="&xacml3;function:anyURI-starts-with">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string"
            >http://example.com/purchase-order/</AttributeValue>
          <AttributeDesignator
            Category="&xacml3;attribute-category:resource"
            AttributeId="&xacml1;resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            MustBePresent="false"/>
        </Match>
```

```
    </AllOf>
  </AnyOf>
</Target>

<!-- Create a bag containing only relevant action history records. -->
<VariableDefinition VariableId="relevant-history">
  <Select VariableId="record">
    <AttributeDesignator
      Category="&xacml3;attribute-category:resource"
      AttributeId="&xacml3;sod:attribute:history"
      DataType="&xacml3;data-type:entity"
      MustBePresent="false"/>
    <Apply FunctionId="&xacml1;function:and">

      <!-- Matching constraint-id. -->
      <Apply FunctionId="&xacml1;function:string-is-in">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string"
          >purchase-order</AttributeValue>
        <!-- Fetch the constraint-id from the action history record. -->
        <Apply FunctionId="&xacml3;function:attribute-designator">
          <VariableReference VariableId="record"/>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >&xacml3;sod:attribute:constraint-id</AttributeValue>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
        </Apply>
      </Apply>

      <!-- Matching transaction-id. -->
      <Apply FunctionId="&xacml1;function:anyURI-at-least-one-member-of">
        <!-- Fetch the transaction-id from the action history record. -->
        <Apply FunctionId="&xacml3;function:attribute-designator">
          <VariableReference VariableId="record"/>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >&xacml3;sod:attribute:transaction-id</AttributeValue>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://www.w3.org/2001/XMLSchema#anyURI</AttributeValue>
        </Apply>
        <!-- The resource-id is used as the transaction-id. -->
        <AttributeDesignator
          Category="&xacml3;attribute-category:resource"
          AttributeId="&xacml1;resource:resource-id"
          DataType="http://www.w3.org/2001/XMLSchema#anyURI"
          MustBePresent="false"/>
      </Apply>

    </Apply>
  </Select>
</VariableDefinition>

<!-- A reusable test that the current action is 'raise'. -->
<VariableDefinition VariableId="action-is-raise">
  <Apply FunctionId="&xacml1;function:string-is-in">
    <AttributeValue
      DataType="http://www.w3.org/2001/XMLSchema#string"
      >raise</AttributeValue>
    <AttributeDesignator
      Category="&xacml3;attribute-category:action"
      AttributeId="&xacml1;action:action-id"
```



```
      DataType="http://www.w3.org/2001/XMLSchema#string"
      MustBePresent="false"/>
  </Apply>
</VariableDefinition>

<!-- A reusable test that the current action is 'approve'. -->
<VariableDefinition VariableId="action-is-approve">
  <Apply FunctionId="&xacml1;function:string-is-in">
    <AttributeValue
      DataType="http://www.w3.org/2001/XMLSchema#string"
      >approve</AttributeValue>
    <AttributeDesignator
      Category="&xacml3;attribute-category:action"
      AttributeId="&xacml1;action:action-id"
      DataType="http://www.w3.org/2001/XMLSchema#string"
      MustBePresent="false"/>
  </Apply>
</VariableDefinition>

<!-- Rules applicable to raising a purchase order. -->

<Rule RuleId="raise-only-once" Effect="Deny">
  <Description>
    Make sure the purchase order hasn't already been raised.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-raise"/>
      <ForAny VariableId="record">
        <VariableReference VariableId="relevant-history"/>
        <Apply FunctionId="&xacml1;function:string-is-in">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string"
            >raise</AttributeValue>
          <Apply FunctionId="&xacml3;function:attribute-designator">
            <VariableReference VariableId="record"/>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >&xacml1;action:action-id</AttributeValue>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
          </Apply>
        </Apply>
      </ForAny>
    </Apply>
  </Condition>
</Rule>

<Rule RuleId="raise-purchase-order" Effect="Permit">
  <Description>
    Allow a purchase order to be raised by any employee.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-raise"/>
      <Apply FunctionId="&xacml1;function:any-of">
        <Function FunctionId="&xacml1;function:rfc822Name-match"/>
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string"
          >example.com</AttributeValue>
        <AttributeDesignator
          Category="&xacml1;subject-category:access-subject"
          AttributeId="&xacml1;subject:subject-id"
```



```
            DataType="&xacml1;data-type:rfc822Name"
            MustBePresent="false"/>
        </Apply>
      </Apply>
    </Condition>
    <ObligationExpressions>
      <!-- Return an action history record for the raise action. -->
      <ObligationExpression ObligationId="&xacml3;sod:obligation:add-history"
        FulfillOn="Permit">

        <AttributeAssignmentExpression
          AttributeId="&xacml1;resource:resource-id">
          <AttributeDesignator
            Category="&xacml3;attribute-category:resource"
            AttributeId="&xacml1;resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml1;subject:subject-id">
          <AttributeDesignator
            Category="&xacml1;subject-category:access-subject"
            AttributeId="&xacml1;subject:subject-id"
            DataType="&xacml1;data-type:rfc822Name"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml1;action:action-id">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >raise</AttributeValue>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml3;sod:attribute:constraint-id">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >purchase-order</AttributeValue>
        </AttributeAssignmentExpression>

        <!-- The resource-id is used as the transaction-id. -->
        <AttributeAssignmentExpression
          AttributeId="&xacml3;sod:attribute:transaction-id">
          <AttributeDesignator
            Category="&xacml3;attribute-category:resource"
            AttributeId="&xacml1;resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

        <!-- Save the raiser's department attribute for
             later use in approval validation. -->
        <AttributeAssignmentExpression
          AttributeId="urn:example:xacml:department">
          <AttributeDesignator
            Category="&xacml1;subject-category:access-subject"
            AttributeId="urn:example:xacml:department"
            DataType="http://www.w3.org/2001/XMLSchema#string"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

      </ObligationExpression>
    </ObligationExpressions>
  </Rule>
```



```
<!-- Rules applicable to approving a purchase order. -->

<Rule RuleId="approve-only-raised" Effect="Deny">
  <Description>
    Make sure the purchase order has been raised.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-approve"/>
      <Apply FunctionId="&xacml1;function:not">
        <ForAny VariableId="record">
          <VariableReference VariableId="relevant-history"/>
          <Apply FunctionId="&xacml1;function:string-is-in">
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#string"
              >raise</AttributeValue>
            <Apply FunctionId="&xacml3;function:attribute-designator">
              <VariableReference VariableId="record"/>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >&xacml1;action:action-id</AttributeValue>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
            </Apply>
          </Apply>
        </ForAny>
      </Apply>
    </Apply>
  </Condition>
</Rule>

<Rule RuleId="approve-only-once" Effect="Deny">
  <Description>
    Make sure the purchase order hasn't already been approved.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-approve"/>
      <ForAny VariableId="record">
        <VariableReference VariableId="relevant-history"/>
        <Apply FunctionId="&xacml1;function:string-is-in">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string"
            >approve</AttributeValue>
          <Apply FunctionId="&xacml3;function:attribute-designator">
            <VariableReference VariableId="record"/>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >&xacml1;action:action-id</AttributeValue>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
          </Apply>
        </Apply>
      </ForAny>
    </Apply>
  </Condition>
</Rule>

<Rule RuleId="not-raised-by-approver" Effect="Deny">
  <Description>
    Make sure the purchase order wasn't raised by the prospective approver.
```



```
    </Description>
    <Condition>
      <Apply FunctionId="&xacml1;function:and">
        <VariableReference VariableId="action-is-approve"/>
        <ForAny VariableId="record">
          <VariableReference VariableId="relevant-history"/>
          <Apply FunctionId="&xacml1;function:and">
            <Apply FunctionId="&xacml1;function:string-is-in">
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#string"
                >raise</AttributeValue>
              <Apply FunctionId="&xacml3;function:attribute-designator">
                <VariableReference VariableId="record"/>
                <AttributeValue
                  DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                  >&xacml1;action:action-id</AttributeValue>
                <AttributeValue
                  DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                  >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
              </Apply>
            </Apply>
            <Apply
              FunctionId="&xacml1;function:rfc822Name-at-least-one-member-of">
              <AttributeDesignator
                Category="&xacml1;subject-category:access-subject"
                AttributeId="&xacml1;subject:subject-id"
                DataType="&xacml1;data-type:rfc822Name"
                MustBePresent="false"/>
              <Apply FunctionId="&xacml3;function:attribute-designator">
                <VariableReference VariableId="record"/>
                <AttributeValue
                  DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                  >&xacml1;subject:subject-id</AttributeValue>
                <AttributeValue
                  DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                  >&xacml1;data-type:rfc822Name</AttributeValue>
              </Apply>
            </Apply>
          </Apply>
        </ForAny>
      </Apply>
    </Condition>
  </Rule>

  <Rule RuleId="approve-purchase-order" Effect="Permit">
    <Description>
      Allow a purchase order to be approved by
      the raiser's department manager.
    </Description>
    <Condition>
      <Apply FunctionId="&xacml1;function:and">
        <VariableReference VariableId="action-is-approve"/>

        <!-- Approver is a department head. -->
        <Apply FunctionId="&xacml1;function:string-is-in">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >Department Head</AttributeValue>
          <AttributeDesignator
            Category="&xacml1;subject-category:access-subject"
            AttributeId="urn:example:xacml:job-title"
            DataType="http://www.w3.org/2001/XMLSchema#string"
            MustBePresent="false"/>
        </Apply>
```



```
        <ForAny VariableId="record">
          <VariableReference VariableId="relevant-history"/>
          <!-- Raiser and approver are in the same department. -->
          <Apply
            FunctionId="&xacml1;function:string-at-least-one-member-of">
            <!-- Fetch approver's department. -->
            <AttributeDesignator
              Category="&xacml1;subject-category:access-subject"
              AttributeId="urn:example:xacml:department"
              DataType="http://www.w3.org/2001/XMLSchema#string"
              MustBePresent="false"/>
            <!-- Fetch raiser's department. -->
            <Apply FunctionId="&xacml3;function:attribute-designator">
              <VariableReference VariableId="record"/>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >urn:example:xacml:department</AttributeValue>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
            </Apply>
          </Apply>
        </ForAny>

      </Apply>
    </Condition>
    <ObligationExpressions>
      <!-- Return an action history record for the approve action. -->
      <ObligationExpression ObligationId="&xacml3;sod:obligation:add-history"
        FulfillOn="Permit">

        <AttributeAssignmentExpression
          AttributeId="&xacml1;resource:resource-id">
          <AttributeDesignator
            Category="&xacml3;attribute-category:resource"
            AttributeId="&xacml1;resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml1;subject:subject-id">
          <AttributeDesignator
            Category="&xacml1;subject-category:access-subject"
            AttributeId="&xacml1;subject:subject-id"
            DataType="&xacml1;data-type:rfc822Name"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml1;action:action-id">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >approve</AttributeValue>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml3;sod:attribute:constraint-id">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >purchase-order</AttributeValue>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml3;sod:attribute:transaction-id">
          <AttributeDesignator
```



```
            Category="&xacml3;attribute-category:resource"
            AttributeId="&xacml1;resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

      </ObligationExpression>
    </ObligationExpressions>
  </Rule>
</Policy>
```

The `raise-only-once` rule prevents a purchase order being raised a second time.

The `raise-purchase-order` rule allows any subject with an email address in the `example.com` domain to perform the `raise` action on a purchase order, and if satisfied, emits an `add-history` obligation to cause the PEP to store an ***action history record*** noting the action and the user performing the action, along with a `constraint-id` and `transaction-id`. The `department` of the subject is also added to be available later for evaluating the `approve` request.

The remaining rules are applicable when the requested action is `approve`.

The `approve-only-raised` rule prevents the approval of a purchase order that hasn't yet been raised.

The `approve-only-once` rule prevents a purchase order being approved a second time.

The `not-raised-by-approver` rule is the principal rule enforcing the SoD constraint that a purchase order must be approved by someone other than the person who raises it. It denies the request if the approver also raised the purchase order.

The `approve-purchase-order` rule allows the purchase order to be approved by an appropriate person, in this case, the head of the department of the person who raised the purchase order.

The `relevant-history` variable is provided for completeness. It ensures that only the ***action history records*** with the appropriate `constraint-id` and `transaction-id` values are considered by the rules. However, since there is only one constraint and the purchase order *is* the transaction, all the ***action history records*** provided by the PEP should match anyway.

8.1.1 Raise Action

Suppose that the subject, Bob, attempts to raise a new purchase order with an authorization request that contains the following attributes:

```
<Request xmlns="&xacml3;core:schema:wd-17"
         ReturnPolicyIdList="false" CombinedDecision="false">
  <Attributes Category="&xacml1;subject-category:access-subject">
    <Attribute AttributeId="&xacml1;subject:subject-id"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml1;data-type:rfc822Name"
        >bob@example.com</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:department"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:job-title"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Accountant</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:action">
    <Attribute AttributeId="&xacml1;action:action-id"
```

```
              IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >raise</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:resource">
    <Attribute AttributeId="&xacml1;resource:resource-id"
              IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/purchase-order/32154</AttributeValue>
    </Attribute>
    <!-- Other attributes detailing the purchase order. -->
    <Attribute AttributeId="urn:example:xacml:description"
              IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >calculator</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:price"
              IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >$30</AttributeValue>
    </Attribute>
  </Attributes>
</Request>
```

The request would be expected to include other categories and attributes in practice, but since the example policy does not reference any such attributes they have been omitted from the example.

The PDP would return the following result from evaluating the request:

```
<Result xmlns="&xacml3;core:schema:wd-17">
  <Decision>Permit</Decision>
  <Status>
   <StatusCode Value="&xacml1;status:ok"/>
  </Status>
  <Obligations>
   <Obligation ObligationId="&xacml3;sod:obligation:add-history">
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
      AttributeId="&xacml1;resource:resource-id"
      >http://example.com/purchase-order/32154</AttributeAssignment>
    <AttributeAssignment DataType="&xacml1;data-type:rfc822Name"
      AttributeId="&xacml1;subject:subject-id"
      >bob@example.com</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml1;action:action-id"
      >raise</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:constraint-id"
      >purchase-order</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
      AttributeId="&xacml3;sod:attribute:transaction-id"
      >http://example.com/purchase-order/32154</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="urn:example:xacml:department"
      >Finance</AttributeAssignment>
   </Obligation>
  </Obligations>
</Result>
```

The example policy is applicable because the `resource-id` attribute value starts with `http://example.com/purchase-order/`.

The `relevant-history` variable evaluates to an empty bag because there is no `history` attribute in the request. Consequently, the `raise-only-once` rule is not applicable.

The `raise-purchase-order` rule evaluates to `Permit` because the action is `raise` and the subject id matches the required email domain. This rule contributes an ***action history record*** to the result in the form of an `add-history` obligation.

The `action-is-approve` variable evaluates to false, so the `approve-only-raised`, `approve-only-once`, `not-raised-by-approver` and `approve-purchase-order` rules are not applicable.

With one rule evaluating to `Permit` and no rule evaluating the `Deny`, the policy evaluates to `Permit` overall. The PEP is obligated to save the ***action history record***.

8.1.2 Unsuccessful Approve Action

Suppose that the subject, Bob, tries to approve the purchase order he raised. The PEP honors the earlier `add-history` obligation by including the ***action history record*** as a value of the `history` attribute in the resource category of the request, as follows:

```
<Request xmlns="&xacml3;core:schema:wd-17"
         ReturnPolicyIdList="false" CombinedDecision="false">
  <Attributes Category="&xacml1;subject-category:access-subject">
    <Attribute AttributeId="&xacml1;subject:subject-id"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml1;data-type:rfc822Name"
        >bob@example.com</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:department"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:job-title"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Accountant</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:action">
    <Attribute AttributeId="&xacml1;action:action-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >approve</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:resource">
    <Attribute AttributeId="&xacml1;resource:resource-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/purchase-order/32154</AttributeValue>
    </Attribute>
    <Attribute AttributeId="&xacml3;sod:attribute:history"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml3;data-type:entity">
        <Attribute AttributeId="&xacml1;resource:resource-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/purchase-order/32154</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;subject:subject-id"
                   IncludeInResult="false">
          <AttributeValue DataType="&xacml1;data-type:rfc822Name"
            >bob@example.com</AttributeValue>
```

```
        </Attribute>
        <Attribute AttributeId="&xacml1;action:action-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >raise</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:constraint-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >purchase-order</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >http://example.com/purchase-order/32154</AttributeValue>
        </Attribute>
        <Attribute AttributeId="urn:example:xacml:department"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >Finance</AttributeValue>
        </Attribute>
      </AttributeValue>
    </Attribute>
  </Attributes>
</Request>
```

The PDP returns the following result from evaluating the request:

```
<Result xmlns="&xacml3;core:schema:wd-17">
  <Decision>Deny</Decision>
  <Status>
    <StatusCode Value="&xacml1;status:ok"/>
  </Status>
</Result>
```

The example policy is applicable because the `resource-id` attribute value starts with `http://example.com/purchase-order/`.

The `action-is-raise` variable evaluates to false, so the `raise-only-once` and `raise-purchase-order` rules are not applicable.

The `relevant-history` variable evaluates to a bag containing the `entity` value from the `history` attribute in the request because the nested `constraint-id` and `transaction-id` attributes match the appropriate values. The `approve-only-raised` rule is not applicable because the `action-id` attribute of that `entity` value has the value `raise` (the condition is satisfied if the `raise` value is *not* found). The `approve-only-once` rule is not applicable because the `approve` action is not found.

The `not-raised-by-approver` rule evaluates to `Deny` because the action is `approve` and the `relevant-history` bag contains an `entity` value where the `action-id` attribute value is `raise` *and* the `subject-id` matches the `subject-id` in the request (i.e., the approver is the same as the raiser).

The policy evaluates to `Deny` overall because of the `not-raised-by-approver` rule, regardless of the `approve-purchase-order` rule (which is not applicable in this case because the subject is not appropriately qualified to approve purchase orders).

8.1.3 Successful Approve Action

Now suppose that the subject, Alice, tries to approve the purchase order raised by Bob. Again, the PEP honors the earlier `add-history` obligation by including the ***action history record*** as a value of the `history` attribute in the resource category of the request, as follows:

```
<Request xmlns="&xacml3;core:schema:wd-17"
         ReturnPolicyIdList="false" CombinedDecision="false">
  <Attributes Category="&xacml1;subject-category:access-subject">
    <Attribute AttributeId="&xacml1;subject:subject-id"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml1;data-type:rfc822Name"
        >alice@example.com</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:department"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:job-title"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Department Head</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:action">
    <Attribute AttributeId="&xacml1;action:action-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >approve</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:resource">
    <Attribute AttributeId="&xacml1;resource:resource-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/purchase-order/32154</AttributeValue>
    </Attribute>
    <Attribute AttributeId="&xacml3;sod:attribute:history"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml3;data-type:entity">
        <Attribute AttributeId="&xacml1;resource:resource-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/purchase-order/32154</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;subject:subject-id"
                   IncludeInResult="false">
          <AttributeValue DataType="&xacml1;data-type:rfc822Name"
            >bob@example.com</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;action:action-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >raise</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:constraint-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >purchase-order</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
                   IncludeInResult="false">
```

```
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/purchase-order/32154</AttributeValue>
        </Attribute>
        <Attribute AttributeId="urn:example:xacml:department"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >Finance</AttributeValue>
        </Attribute>
      </AttributeValue>
    </Attribute>
  </Attributes>
</Request>
```

The PDP returns the following result from evaluating the request:

```
<Result xmlns="&xacml3;core:schema:wd-17">
  <Decision>Permit</Decision>
  <Status>
    <StatusCode Value="&xacml1;status:ok"/>
  </Status>
  <Obligations>
    <Obligation ObligationId="&xacml3;sod:obligation:add-history">
      <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        AttributeId="&xacml1;resource:resource-id"
        >http://example.com/purchase-order/32154</AttributeAssignment>
      <AttributeAssignment DataType="&xacml1;data-type:rfc822Name"
        AttributeId="&xacml1;subject:subject-id"
        >alice@example.com</AttributeAssignment>
      <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
        AttributeId="&xacml1;action:action-id"
        >approve</AttributeAssignment>
      <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
        AttributeId="&xacml3;sod:attribute:constraint-id"
        >purchase-order</AttributeAssignment>
      <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        AttributeId="&xacml3;sod:attribute:transaction-id"
        >http://example.com/purchase-order/32154</AttributeAssignment>
    </Obligation>
  </Obligations>
</Result>
```

The example policy is applicable because the `resource-id` attribute value starts with `http://example.com/purchase-order/`.

The `action-is-raise` variable evaluates to false, so the `raise-only-once` and `raise-purchase-order` rules are not applicable.

The `relevant-history` variable evaluates to a bag containing the `entity` value from the `history` attribute in the request because the nested `constraint-id` and `transaction-id` attributes match the appropriate values. The `approve-only-raised` rule is not applicable because the `action-id` attribute of that `entity` value has the value `raise`. The `approve-only-once` rule is not applicable because the `approve` action is not found.

The `not-raised-by-approver` rule is not applicable. Although the action is `approve`, the `relevant-history` bag does not contain an `entity` value where the `action-id` attribute value is `raise` and the `subject-id` matches the `subject-id` in the request. So, the approver is not the same as the raiser.

The `approve-purchase-order` rule evaluates to `Permit` because the `job-title` of the subject is `Department Head` and the `relevant-history` bag contains an `entity` value where the

`department` attribute value matches the subject's department. This rule contributes an ***action history record*** to the result in the form of an `add-history` obligation indicating that Alice approved the purchase order. The policy evaluates to `Permit` overall.

This example, as far as it goes, does not make use of the `end-history` obligation or the `time-limit` attribute, so the PEP is required to retain the two ***action history records*** until the resource (i.e., the purchase order) is removed. In practice, there may be further policy rules to address additional authorization steps in the purchase workflow, e.g., for accepting delivery, invoicing and payment, which may include the `end-history` obligation. Or the PEP may retain the purchase order and ***action history records*** indefinitely (perhaps in a reduced or more concise form) for archiving or auditing purposes.

8.2 Account Deduction Example

This example shows a policy for enforcing an SoD constraint that a deduction from a persistent cash account must be approved by someone other than the person who requests it. In addition, the example supports multiple deduction requests being progressed simultaneously as separate transactions.

An account is represented as a resource with the following attributes:

- `urn:oasis:names:tc:xacml:1.0:resource:resource-id`

  A unique URI identifying the account.

- `urn:oasis:names:tc:xacml:3.0:sod:attribute:history`

  The current collection of ***action history records*** for the account, possibly pertaining to multiple transactions.

- Other attributes detailing the account but not referenced by the policy such as the current account balance.

An employee is represented as a subject with the following attributes:

- `urn:oasis:names:tc:xacml:1.0:subject:subject-id`

  A unique employee identifier in the form of an email address.

- `urn:example:xacml:department`

  The department to which the employee is assigned.

- `urn:example:xacml:job-title`

  The title of the job performed by the employee.

- Other attributes detailing the employee but not referenced by the policy.

A cash account can be assumed to be subjected to many independent deposits and withdrawals over its lifetime and some of those operations will overlap in time. The `transaction-id` attribute will be required to separate ***action history records*** belonging to different operations happening around the same time. In this example a unique transaction identifier is generated by the PDP using the `get-string-identifier` function when a withdrawal is requested. The application requesting an authorization decision might already have a unique identifier for the transaction that it can include as a `transaction-id` attribute in the action category of the authorization request. In this case the policy would be rewritten to use and copy the `transaction-id` from the action category instead of generating a value.

This example only deals with approval for withdrawals from the account and covers only one SoD constraint identified with the string constant `withdrawal`. In practice we might expect that there are other operations on the account that are covered by other SoD constraints with their own distinct identifiers.

```
<Policy xmlns="&xacml3;core:schema:wd-17"
  PolicyId="http://example.com/SoD/accounts" Version="1.0"
  RuleCombiningAlgId="&xacml3;rule-combining-algorithm:deny-overrides">

  <Description>
    Policy for SoD constraints applicable to withdrawing funds from
```

```
  a financial account.
</Description>

<!-- The target restricts applicability to account resources. -->
<Target>
  <AnyOf>
    <AllOf>
      <Match
        MatchId="&xacml3;function:anyURI-starts-with">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string"
          >http://example.com/account/</AttributeValue>
        <AttributeDesignator
          Category="&xacml3;attribute-category:resource"
          AttributeId="&xacml1;resource:resource-id"
          DataType="http://www.w3.org/2001/XMLSchema#anyURI"
          MustBePresent="false"/>
      </Match>
    </AllOf>
  </AnyOf>
</Target>

<!-- Create a bag containing only relevant action history records. -->
<VariableDefinition VariableId="relevant-history">
  <Select VariableId="record">
    <AttributeDesignator
      Category="&xacml3;attribute-category:resource"
      AttributeId="&xacml3;sod:attribute:history"
      DataType="&xacml3;data-type:entity"
      MustBePresent="false"/>
    <Apply FunctionId="&xacml1;function:and">

      <!-- Matching resource-id. -->
      <Apply FunctionId="&xacml1;function:anyURI-at-least-one-member-of">
        <!-- Fetch the resource-id from the action history record. -->
        <Apply FunctionId="&xacml3;function:attribute-designator">
          <VariableReference VariableId="record"/>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >&xacml1;resource:resource-id</AttributeValue>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://www.w3.org/2001/XMLSchema#anyURI</AttributeValue>
        </Apply>
        <AttributeDesignator
          Category="&xacml3;attribute-category:action"
          AttributeId="&xacml1;resource:resource-id"
          DataType="http://www.w3.org/2001/XMLSchema#anyURI"
          MustBePresent="false"/>
      </Apply>

      <!-- Matching constraint-id. -->
      <Apply FunctionId="&xacml1;function:string-is-in">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string"
          >withdrawal</AttributeValue>
        <!-- Fetch the constraint-id from the action history record. -->
        <Apply FunctionId="&xacml3;function:attribute-designator">
          <VariableReference VariableId="record"/>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >&xacml3;sod:attribute:constraint-id</AttributeValue>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
```



```
            >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
        </Apply>
      </Apply>

      <!-- Matching transaction-id. -->
      <Apply FunctionId="&xacml1;function:string-at-least-one-member-of">
        <!-- Fetch the transaction-id from the action history record. -->
        <Apply FunctionId="&xacml3;function:attribute-designator">
          <VariableReference VariableId="record"/>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >&xacml3;sod:attribute:transaction-id</AttributeValue>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
        </Apply>
        <AttributeDesignator
          Category="&xacml3;attribute-category:action"
          AttributeId="&xacml3;sod:attribute:transaction-id"
          DataType="http://www.w3.org/2001/XMLSchema#string"
          MustBePresent="false"/>
      </Apply>

    </Apply>
  </Select>
</VariableDefinition>

<!-- A reusable test that the current action is 'request-withdrawal'. -->
<VariableDefinition VariableId="action-is-request-withdrawal">
  <Apply FunctionId="&xacml1;function:string-is-in">
    <AttributeValue
      DataType="http://www.w3.org/2001/XMLSchema#string"
      >request-withdrawal</AttributeValue>
    <AttributeDesignator
      Category="&xacml3;attribute-category:action"
      AttributeId="&xacml1;action:action-id"
      DataType="http://www.w3.org/2001/XMLSchema#string"
      MustBePresent="false"/>
  </Apply>
</VariableDefinition>

<!-- A reusable test that the current action is 'approve'. -->
<VariableDefinition VariableId="action-is-approve">
  <Apply FunctionId="&xacml1;function:string-is-in">
    <AttributeValue
      DataType="http://www.w3.org/2001/XMLSchema#string"
      >approve</AttributeValue>
    <AttributeDesignator
      Category="&xacml3;attribute-category:action"
      AttributeId="&xacml1;action:action-id"
      DataType="http://www.w3.org/2001/XMLSchema#string"
      MustBePresent="false"/>
  </Apply>
</VariableDefinition>

<!-- A reusable test that the subject is an accountant
     in the Finance Department. -->
<VariableDefinition VariableId="accountant-in-finance">
  <Apply FunctionId="&xacml1;function:and">

    <!-- Subject is an accountant. -->
    <Apply FunctionId="&xacml1;function:string-is-in">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Accountant</AttributeValue>
```

```
      <AttributeDesignator
        Category="&xacml1;subject-category:access-subject"
        AttributeId="urn:example:xacml:job-title"
        DataType="http://www.w3.org/2001/XMLSchema#string"
        MustBePresent="false"/>
    </Apply>

    <!-- Subject works in the Finance Department. -->
    <Apply FunctionId="&xacml1;function:string-is-in">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
      <AttributeDesignator
        Category="&xacml1;subject-category:access-subject"
        AttributeId="urn:example:xacml:department"
        DataType="http://www.w3.org/2001/XMLSchema#string"
        MustBePresent="false"/>
    </Apply>

  </Apply>
</VariableDefinition>

<!-- Rules applicable to requesting a withdrawal from an account. -->

<Rule RuleId="request-only-once" Effect="Deny">
  <Description>
    Make sure the transaction has a single withdrawal request.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-request-withdrawal"/>
      <ForAny VariableId="record">
        <VariableReference VariableId="relevant-history"/>
        <Apply FunctionId="&xacml1;function:string-is-in">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string"
            >request-withdrawal</AttributeValue>
          <Apply FunctionId="&xacml3;function:attribute-designator">
            <VariableReference VariableId="record"/>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >&xacml1;action:action-id</AttributeValue>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
          </Apply>
        </Apply>
      </ForAny>
    </Apply>
  </Condition>
</Rule>

<Rule RuleId="request-withdrawal" Effect="Permit">
  <Description>
    Allow a withdrawal to be requested by any accountant in
    the Finance Department.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-request-withdrawal"/>
      <VariableReference VariableId="accountant-in-finance"/>
    </Apply>
  </Condition>
  <ObligationExpressions>
    <!-- Return an action history record for
```



```
          the request-withdrawal action. -->
      <ObligationExpression ObligationId="&xacml3;sod:obligation:add-history"
        FulfillOn="Permit">

        <AttributeAssignmentExpression
          AttributeId="&xacml1;resource:resource-id">
          <AttributeDesignator
            Category="&xacml3;attribute-category:resource"
            AttributeId="&xacml1;resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml1;subject:subject-id">
          <AttributeDesignator
            Category="&xacml1;subject-category:access-subject"
            AttributeId="&xacml1;subject:subject-id"
            DataType="&xacml1;data-type:rfc822Name"
            MustBePresent="false"/>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml1;action:action-id">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >request-withdrawal</AttributeValue>
        </AttributeAssignmentExpression>

        <AttributeAssignmentExpression
          AttributeId="&xacml3;sod:attribute:constraint-id">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >withdrawal</AttributeValue>
        </AttributeAssignmentExpression>

        <!-- Each withdrawal request is distinct
             and is given a unique identifier. -->
        <AttributeAssignmentExpression
          AttributeId="&xacml3;sod:attribute:transaction-id">
          <Apply FunctionId="&xacml3;function:get-string-identifier"/>
        </AttributeAssignmentExpression>

        <!-- Set the time limit to three days from now. -->
        <AttributeAssignmentExpression
          AttributeId="&xacml3;sod:attribute:time-limit">
          <Apply FunctionId="&xacml3;function:dateTime-add-dayTimeDuration">
            <Apply FunctionId="&xacml1;function:dateTime-one-and-only">
              <AttributeDesignator
                Category="&xacml3;attribute-category:environment"
                AttributeId="&xacml1;environment:current-dateTime"
                DataType="http://www.w3.org/2001/XMLSchema#dateTime"
                MustBePresent="false"/>
            </Apply>
            <AttributeValue DataType="&xacml2;data-type:dayTimeDuration"
              >P3D</AttributeValue>
          </Apply>
        </AttributeAssignmentExpression>

      </ObligationExpression>
    </ObligationExpressions>
  </Rule>

  <!-- Rules applicable to approving a withdrawal. -->

  <Rule RuleId="approve-only-requested" Effect="Deny">
```



```
  <Description>
    Make sure the withdrawal has been requested.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-approve"/>
      <Apply FunctionId="&xacml1;function:not">
        <ForAny VariableId="record">
          <VariableReference VariableId="relevant-history"/>
          <Apply FunctionId="&xacml1;function:string-is-in">
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#string"
              >request-withdrawal</AttributeValue>
            <Apply FunctionId="&xacml3;function:attribute-designator">
              <VariableReference VariableId="record"/>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >&xacml1;action:action-id</AttributeValue>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
            </Apply>
          </Apply>
        </ForAny>
      </Apply>
    </Apply>
  </Condition>
</Rule>

<Rule RuleId="approve-only-once" Effect="Deny">
  <Description>
    Make sure the withdrawal hasn't already been approved.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-approve"/>
      <ForAny VariableId="record">
        <VariableReference VariableId="relevant-history"/>
        <Apply FunctionId="&xacml1;function:string-is-in">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string"
            >approve</AttributeValue>
          <Apply FunctionId="&xacml3;function:attribute-designator">
            <VariableReference VariableId="record"/>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >&xacml1;action:action-id</AttributeValue>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
          </Apply>
        </Apply>
      </ForAny>
    </Apply>
  </Condition>
</Rule>

<Rule RuleId="not-requested-by-approver" Effect="Deny">
  <Description>
    Make sure the withdrawal wasn't requested by the prospective approver.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-approve"/>
```



```
      <ForAny VariableId="record">
        <VariableReference VariableId="relevant-history"/>
        <Apply FunctionId="&xacml1;function:and">
          <Apply FunctionId="&xacml1;function:string-is-in">
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#string"
              >request-withdrawal</AttributeValue>
            <Apply FunctionId="&xacml3;function:attribute-designator">
              <VariableReference VariableId="record"/>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >&xacml1;action:action-id</AttributeValue>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
            </Apply>
          </Apply>
          <Apply
            FunctionId="&xacml1;function:rfc822Name-at-least-one-member-of">
            <AttributeDesignator
              Category="&xacml1;subject-category:access-subject"
              AttributeId="&xacml1;subject:subject-id"
              DataType="&xacml1;data-type:rfc822Name"
              MustBePresent="false"/>
            <Apply FunctionId="&xacml3;function:attribute-designator">
              <VariableReference VariableId="record"/>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >&xacml1;subject:subject-id</AttributeValue>
              <AttributeValue
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"
                >&xacml1;data-type:rfc822Name</AttributeValue>
            </Apply>
          </Apply>
        </Apply>
      </ForAny>
    </Apply>
  </Condition>
</Rule>

<Rule RuleId="approve-withdrawal" Effect="Permit">
  <Description>
    Allow a withdrawal to be approved by any (other) accountant
    in the Finance Department.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">
      <VariableReference VariableId="action-is-approve"/>
      <VariableReference VariableId="accountant-in-finance"/>
    </Apply>
  </Condition>
  <ObligationExpressions>
    <!-- Return an action history record for the approve action. -->
    <ObligationExpression ObligationId="&xacml3;sod:obligation:add-history"
      FulfillOn="Permit">

      <AttributeAssignmentExpression
        AttributeId="&xacml1;resource:resource-id">
        <AttributeDesignator
          Category="&xacml3;attribute-category:resource"
          AttributeId="&xacml1;resource:resource-id"
          DataType="http://www.w3.org/2001/XMLSchema#anyURI"
          MustBePresent="false"/>
      </AttributeAssignmentExpression>
```



```
      <AttributeAssignmentExpression
        AttributeId="&xacml1;subject:subject-id">
        <AttributeDesignator
          Category="&xacml1;subject-category:access-subject"
          AttributeId="&xacml1;subject:subject-id"
          DataType="&xacml1;data-type:rfc822Name"
          MustBePresent="false"/>
      </AttributeAssignmentExpression>

      <AttributeAssignmentExpression
        AttributeId="&xacml1;action:action-id">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
          >approve</AttributeValue>
      </AttributeAssignmentExpression>

      <AttributeAssignmentExpression
        AttributeId="&xacml3;sod:attribute:constraint-id">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
          >withdrawal</AttributeValue>
      </AttributeAssignmentExpression>

      <AttributeAssignmentExpression
        AttributeId="&xacml3;sod:attribute:transaction-id">
        <AttributeDesignator
          Category="&xacml3;attribute-category:action"
          AttributeId="&xacml3;sod:attribute:transaction-id"
          DataType="http://www.w3.org/2001/XMLSchema#string"
          MustBePresent="false"/>
      </AttributeAssignmentExpression>

      <!-- Set the time limit to three days from now. -->
      <AttributeAssignmentExpression
        AttributeId="&xacml3;sod:attribute:time-limit">
        <Apply FunctionId="&xacml3;function:dateTime-add-dayTimeDuration">
          <Apply FunctionId="&xacml1;function:dateTime-one-and-only">
            <AttributeDesignator
              Category="&xacml3;attribute-category:environment"
              AttributeId="&xacml1;environment:current-dateTime"
              DataType="http://www.w3.org/2001/XMLSchema#dateTime"
              MustBePresent="false"/>
          </Apply>
          <AttributeValue
            DataType="&xacml2;data-type:dayTimeDuration"
            >P3D</AttributeValue>
        </Apply>
      </AttributeAssignmentExpression>

    </ObligationExpression>
  </ObligationExpressions>
</Rule>

<Rule RuleId="make-withdrawal" Effect="Permit">
  <Description>
    Allow any accountant in the Finance Department to initiate
    an approved withdrawal.
  </Description>
  <Condition>
    <Apply FunctionId="&xacml1;function:and">

      <!-- Action is 'withdraw'. -->
      <Apply FunctionId="&xacml1;function:string-is-in">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string"
```



```
          >withdraw</AttributeValue>
        <AttributeDesignator
          Category="&xacml3;attribute-category:action"
          AttributeId="&xacml1;action:action-id"
          DataType="http://www.w3.org/2001/XMLSchema#string"
          MustBePresent="false"/>
      </Apply>

      <VariableReference VariableId="accountant-in-finance"/>

      <!-- Withdrawal is approved. -->
      <ForAny VariableId="record">
        <VariableReference VariableId="relevant-history"/>
        <Apply FunctionId="&xacml1;function:string-is-in">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string"
            >approve</AttributeValue>
          <Apply FunctionId="&xacml3;function:attribute-designator">
            <VariableReference VariableId="record"/>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >&xacml1;action:action-id</AttributeValue>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#anyURI"
              >http://www.w3.org/2001/XMLSchema#string</AttributeValue>
          </Apply>
        </Apply>
      </ForAny>

    </Apply>
  </Condition>
  <ObligationExpressions>
    <!-- Transaction is finished; clean up the action history records. -->
    <ObligationExpression ObligationId="&xacml3;sod:obligation:end-history"
      FulfillOn="Permit">

      <AttributeAssignmentExpression
        AttributeId="&xacml1;resource:resource-id">
        <AttributeDesignator
          Category="&xacml3;attribute-category:resource"
          AttributeId="&xacml1;resource:resource-id"
          DataType="http://www.w3.org/2001/XMLSchema#anyURI"
          MustBePresent="false"/>
      </AttributeAssignmentExpression>

      <AttributeAssignmentExpression
        AttributeId="&xacml3;sod:attribute:constraint-id">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
          >withdrawal</AttributeValue>
      </AttributeAssignmentExpression>

      <AttributeAssignmentExpression
        AttributeId="&xacml3;sod:attribute:transaction-id">
        <AttributeDesignator
          Category="&xacml3;attribute-category:action"
          AttributeId="&xacml3;sod:attribute:transaction-id"
          DataType="http://www.w3.org/2001/XMLSchema#string"
          MustBePresent="false"/>
      </AttributeAssignmentExpression>

    </ObligationExpression>
  </ObligationExpressions>
</Rule>
```



```
</Policy>
```

The `request-only-once` rule prevents a second `request-withdrawal` action for an existing transaction.

The `request-withdrawal` rule allows any subject with a job title of `Accountant` who is working in the Finance Department to perform the `request-withdrawal` action on an account, and if satisfied, emits an `add-history` obligation to cause the PEP to store an ***action history record*** noting the action and the user performing the action, along with the `constraint-id` and a generated `transaction-id`. A `time-limit` attribute is also added requiring the transaction to be approved within three days.

The next group of rules are applicable when the requested action is `approve`.

The `approve-only-requested` rule prevents the approval of a withdrawal that hasn't been requested.

The `approve-only-once` rule prevents a withdrawal being approved a second time.

The `not-requested-by-approver` rule is the principal rule enforcing the SoD constraint that a withdrawal must be approved by someone other than the person who requested it. It denies the request if the approver also requested the withdrawal.

The `approve-withdrawal` rule allows the withdrawal to be approved by an appropriate person, in this case, any accountant in the Finance Department.

The `make-withdrawal` rule is applicable when the requested action is `withdraw`. It authorizes the actual deduction from the account and can be actioned by any accountant in the Finance Department for an approved withdrawal.

The `relevant-history` variable ensures that only the ***action history records*** with the appropriate `constraint-id` and `transaction-id` values are considered by the rules.

8.2.1 Withdrawal Request Action

Suppose that the subject, Carol, requests a withdrawal from the payroll account at 2022-10-10T12:00:00Z with an authorization request that contains the following attributes:

```
<Request xmlns="&xacml3;core:schema:wd-17"
         ReturnPolicyIdList="false" CombinedDecision="false">
  <Attributes Category="&xacml1;subject-category:access-subject">
    <Attribute AttributeId="&xacml1;subject:subject-id"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml1;data-type:rfc822Name"
        >carol@example.com</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:department"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:job-title"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Accountant</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:action">
    <Attribute AttributeId="&xacml1;action:action-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >request-withdrawal</AttributeValue>
    </Attribute>
```

```
    <Attribute AttributeId="urn:example:xacml:amount"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#double"
        >87456.00</AttributeValue>
    </Attribute>
    <!-- Other attributes detailing the withdrawal. -->
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:resource">
    <Attribute AttributeId="&xacml1;resource:resource-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/account/payroll</AttributeValue>
    </Attribute>
  </Attributes>
</Request>
```

The request would be expected to include other categories and attributes in practice, but since the example policy does not reference any such attributes they have been omitted from the example.

The PDP would return the following result from evaluating the request:

```
<Result xmlns="&xacml3;core:schema:wd-17">
  <Decision>Permit</Decision>
  <Status>
   <StatusCode Value="&xacml1;status:ok"/>
  </Status>
  <Obligations>
   <Obligation ObligationId="&xacml3;sod:obligation:add-history">
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
      AttributeId="&xacml1;resource:resource-id"
      >http://example.com/account/payroll</AttributeAssignment>
    <AttributeAssignment DataType="&xacml1;data-type:rfc822Name"
      AttributeId="&xacml1;subject:subject-id"
      >carol@example.com</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml1;action:action-id"
      >request-withdrawal</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:constraint-id"
      >withdrawal</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:transaction-id"
      >61b9081d-92f1-46af-aa81-4f8454877619</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#dateTime"
      AttributeId="&xacml3;sod:attribute:time-limit"
      >2022-10-13T12:00:00Z</AttributeAssignment>
   </Obligation>
  </Obligations>
</Result>
```

The example policy is applicable because the `resource-id` attribute value starts with `http://example.com/account/`.

The `relevant-history` variable evaluates to an empty bag because there is no `history` attribute in the request. Consequently, the `request-only-once` rule is not applicable.

The `request-withdrawal` rule evaluates to `Permit` because the action is `request-withdrawal` and the subject is an accountant in the Finance Department. This rule contributes an ***action history record*** to the result in the form of an `add-history` obligation.

The `action-is-approve` variable evaluates to false, so the `approve-only-requested`, `approve-only-once`, `not-requested-by-approver` and `approve-withdrawal` rules are not applicable.

The action is not `withdraw` so the `make-withdrawal` rule is not applicable.

With one rule evaluating to `Permit` and no rule evaluating the `Deny`, the policy evaluates to `Permit` overall. The PEP is obligated to save the ***action history record***. The PEP must additionally include the `transaction-id` attribute from the ***action history record*** in the action category of future requests for the same transaction to progress the transaction any further. If the `transaction-id` is omitted from a subsequent `approve` action then the request will fail because the prerequisite `request-withdrawal` action in the ***action history records*** won't be matched.

The ***action history record*** contains a `time-limit` attribute. If there the next step for this transaction is not taken before `2022-10-13T12:15:00Z` then the PEP can throw away the associated ***action history record***.

8.2.2 Contemporaneous Withdrawal Request Action

Suppose that fifteen minutes later Dave requests a different withdrawal from the same payroll account with an authorization request containing the following attributes:

```
<Request xmlns="&xacml3;core:schema:wd-17"
         ReturnPolicyIdList="false" CombinedDecision="false">
  <Attributes Category="&xacml1;subject-category:access-subject">
    <Attribute AttributeId="&xacml1;subject:subject-id"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml1;data-type:rfc822Name"
        >dave@example.com</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:department"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:job-title"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Accountant</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:action">
    <Attribute AttributeId="&xacml1;action:action-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >request-withdrawal</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:amount"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#double"
        >91345.00</AttributeValue>
    </Attribute>
    <!-- Other attributes detailing the withdrawal. -->
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:resource">
    <Attribute AttributeId="&xacml1;resource:resource-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/account/payroll</AttributeValue>
    </Attribute>
  </Attributes>
</Request>
```

The PDP would return the following result from evaluating the request:

```
<Result xmlns="&xacml3;core:schema:wd-17">
  <Decision>Permit</Decision>
  <Status>
   <StatusCode Value="&xacml1;status:ok"/>
  </Status>
  <Obligations>
   <Obligation ObligationId="&xacml3;sod:obligation:add-history">
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
      AttributeId="&xacml1;resource:resource-id"
      >http://example.com/account/payroll</AttributeAssignment>
    <AttributeAssignment DataType="&xacml1;data-type:rfc822Name"
      AttributeId="&xacml1;subject:subject-id"
      >dave@example.com</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml1;action:action-id"
      >request-withdrawal</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:constraint-id"
      >withdrawal</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:transaction-id"
      >28f44b05-218f-4a4f-9201-044634b6b0fc</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#dateTime"
      AttributeId="&xacml3;sod:attribute:time-limit"
      >2022-10-13T12:15:00Z</AttributeAssignment>
   </Obligation>
  </Obligations>
</Result>
```

The evaluation of the example policy is essentially the same as for Carol's earlier request except that a different `transaction-id` value is returned in the `add-history` obligation. This enables the two transactions to be kept separate.

Dave's transaction won't be progressed any further in this example. If there is no further activity for this transaction by `2022-10-13T12:15:00Z` then the PEP can throw away the associated ***action history record***.

8.2.3 Approve Action

Now suppose that the subject, Bob, tries to approve the withdrawal requested by Carol and now identified as transaction `61b9081d-92f1-46af-aa81-4f8454877619`. The transaction identifier is provided as the value of the `transaction-id` attribute in the action category. The PEP honors the earlier `add-history` obligations from Carol's and Dave's requests by including the ***action history records*** as values of the `history` attribute in the resource category of the request. Note that the PEP selects the ***action history records*** it provides based on their `resource-id` and `time-limit` attribute values. It does not do any additional filtering. The relevance of each provided ***action history record*** is determined by the policies evaluated by the PDP. Although the transactions are independent as far as this example is concerned, this specification allows for transactions and constraints that have interdependencies.

The following request is submitted at `2022-10-11T14:30:00Z`:

```
<Request xmlns="&xacml3;core:schema:wd-17"
         ReturnPolicyIdList="false" CombinedDecision="false">
  <Attributes Category="&xacml1;subject-category:access-subject">
    <Attribute AttributeId="&xacml1;subject:subject-id"
               IncludeInResult="false">
```

```
      <AttributeValue DataType="&xacml1;data-type:rfc822Name"
        >bob@example.com</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:department"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:job-title"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Accountant</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:action">
    <Attribute AttributeId="&xacml1;action:action-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >approve</AttributeValue>
    </Attribute>
    <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >61b9081d-92f1-46af-aa81-4f8454877619</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:resource">
    <Attribute AttributeId="&xacml1;resource:resource-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/account/payroll</AttributeValue>
    </Attribute>
    <Attribute AttributeId="&xacml3;sod:attribute:history"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml3;data-type:entity">
        <Attribute AttributeId="&xacml1;resource:resource-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/account/payroll</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;subject:subject-id"
                   IncludeInResult="false">
          <AttributeValue DataType="&xacml1;data-type:rfc822Name"
            >carol@example.com</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;action:action-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >request-withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:constraint-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >61b9081d-92f1-46af-aa81-4f8454877619</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:time-limit"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#dateTime"
            >2022-10-13T12:00:00Z</AttributeValue>
```



```
        </Attribute>
      </AttributeValue>
      <AttributeValue DataType="&xacml3;data-type:entity">
        <Attribute AttributeId="&xacml1;resource:resource-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/account/payroll</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;subject:subject-id"
                   IncludeInResult="false">
          <AttributeValue DataType="&xacml1;data-type:rfc822Name"
            >dave@example.com</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;action:action-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >request-withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:constraint-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >28f44b05-218f-4a4f-9201-044634b6b0fc</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:time-limit"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#dateTime"
            >2022-10-13T12:15:00Z</AttributeValue>
        </Attribute>
      </AttributeValue>
    </Attribute>
  </Attributes>
</Request>
```

The PDP returns the following result from evaluating the request:

```
<Result xmlns="&xacml3;core:schema:wd-17">
  <Decision>Permit</Decision>
  <Status>
    <StatusCode Value="&xacml1;status:ok"/>
  </Status>
  <Obligations>
   <Obligation ObligationId="&xacml3;sod:obligation:add-history">
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
      AttributeId="&xacml1;resource:resource-id"
      >http://example.com/account/payroll</AttributeAssignment>
    <AttributeAssignment DataType="&xacml1;data-type:rfc822Name"
      AttributeId="&xacml1;subject:subject-id"
      >bob@example.com</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml1;action:action-id"
      >approve</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:constraint-id"
      >withdrawal</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:transaction-id"
      >61b9081d-92f1-46af-aa81-4f8454877619</AttributeAssignment>
```



```
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#dateTime"
      AttributeId="&xacml3;sod:attribute:time-limit"
      >2022-10-14T14:30:00Z</AttributeAssignment>
   </Obligation>
  </Obligations>
</Result>
```

The example policy is applicable because the `resource-id` attribute value starts with `http://example.com/payroll/`.

The `action-is-request-withdrawal` variable evaluates to false, so the `request-only-once` and `request-withdrawal` rules are not applicable.

The `relevant-history` variable evaluates to a bag containing the `entity` value from the `history` attribute with `transaction-id` equal to `61b9081d-92f1-46af-aa81-4f8454877619`, i.e., Carol's request. Dave's request doesn't match the `transaction-id`. The `approve-only-requested` rule is not applicable because the `action-id` attribute of that `entity` value has the value `request-withdrawal`. The `approve-only-once` rule is not applicable because the `approve` action is not found. The `not-raised-by-approver` rule is not applicable because `bob@example.com` did not perform the `request-withdrawal` action.

The `approve-withdrawal` rule evaluates to `Permit` because the `action-id` is `approve`, the `job-title` of the subject is `Accountant` and the `department` is Finance. This rule contributes an ***action history record*** to the result in the form of an `add-history` obligation indicating that Bob approved the withdrawal. The policy evaluates to `Permit` overall.

The ***action history record*** contains a `time-limit` attribute which effectively extends the time limit for the transaction to `2022-10-14T14:30:00Z`.

8.2.4 Withdraw Action

Carol's withdrawal request is approved so the next step is to make the actual withdrawal, which can be invoked by any accountant in the Finance Department. We'll assume Carol does it. The action is `make-withdrawal`. The transaction identifier is provided as the value of the `transaction-id` attribute in the action category. The PEP honors the earlier `add-history` obligations from Carol's and Dave's requests and Bob's approval by including the ***action history records*** as values of the `history` attribute in the resource category of the request.

The following request is submitted at `2022-10-11T15:00:00Z`, well within the time limit:

```
<Request xmlns="&xacml3;core:schema:wd-17"
         ReturnPolicyIdList="false" CombinedDecision="false">
  <Attributes Category="&xacml1;subject-category:access-subject">
    <Attribute AttributeId="&xacml1;subject:subject-id"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml1;data-type:rfc822Name"
        >carol@example.com</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:department"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Finance</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:example:xacml:job-title"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >Accountant</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:action">
```

```
    <Attribute AttributeId="&xacml1;action:action-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >withdraw</AttributeValue>
    </Attribute>
    <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
        >61b9081d-92f1-46af-aa81-4f8454877619</AttributeValue>
    </Attribute>
  </Attributes>
  <Attributes Category="&xacml3;attribute-category:resource">
    <Attribute AttributeId="&xacml1;resource:resource-id"
               IncludeInResult="false">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
        >http://example.com/account/payroll</AttributeValue>
    </Attribute>
    <Attribute AttributeId="&xacml3;sod:attribute:history"
               IncludeInResult="false">
      <AttributeValue DataType="&xacml3;data-type:entity">
        <Attribute AttributeId="&xacml1;resource:resource-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/account/payroll</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;subject:subject-id"
                   IncludeInResult="false">
          <AttributeValue DataType="&xacml1;data-type:rfc822Name"
            >carol@example.com</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;action:action-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >request-withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:constraint-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >61b9081d-92f1-46af-aa81-4f8454877619</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:time-limit"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#dateTime"
            >2022-10-13T12:00:00Z</AttributeValue>
        </Attribute>
      </AttributeValue>
      <AttributeValue DataType="&xacml3;data-type:entity">
        <Attribute AttributeId="&xacml1;resource:resource-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/account/payroll</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;subject:subject-id"
                   IncludeInResult="false">
          <AttributeValue DataType="&xacml1;data-type:rfc822Name"
            >dave@example.com</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;action:action-id"
                   IncludeInResult="false">
```



```
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >request-withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:constraint-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >28f44b05-218f-4a4f-9201-044634b6b0fc</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:time-limit"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#dateTime"
            >2022-10-13T12:15:00Z</AttributeValue>
        </Attribute>
      </AttributeValue>
      <AttributeValue DataType="&xacml3;data-type:entity">
        <Attribute AttributeId="&xacml1;resource:resource-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI"
            >http://example.com/account/payroll</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;subject:subject-id"
                   IncludeInResult="false">
          <AttributeValue DataType="&xacml1;data-type:rfc822Name"
            >bob@example.com</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml1;action:action-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >approve</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:constraint-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >withdrawal</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:transaction-id"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"
            >61b9081d-92f1-46af-aa81-4f8454877619</AttributeValue>
        </Attribute>
        <Attribute AttributeId="&xacml3;sod:attribute:time-limit"
                   IncludeInResult="false">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#dateTime"
            >2022-10-14T14:30:00Z</AttributeValue>
        </Attribute>
      </AttributeValue>
    </Attribute>
  </Attributes>
</Request>
```

The PDP returns the following result from evaluating the request:

```
<Result xmlns="&xacml3;core:schema:wd-17">
  <Decision>Permit</Decision>
  <Status>
    <StatusCode Value="&xacml1;status:ok"/>
  </Status>
```



```
  <Obligations>
   <Obligation ObligationId="&xacml3;sod:obligation:end-history">
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#anyURI"
      AttributeId="&xacml1;resource:resource-id"
      >http://example.com/account/payroll</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:constraint-id"
      >withdrawal</AttributeAssignment>
    <AttributeAssignment DataType="http://www.w3.org/2001/XMLSchema#string"
      AttributeId="&xacml3;sod:attribute:transaction-id"
      >61b9081d-92f1-46af-aa81-4f8454877619</AttributeAssignment>
   </Obligation>
  </Obligations>
</Result>
```

The example policy is applicable because the `resource-id` attribute value starts with `http://example.com/payroll/`.

The `action-is-request-withdrawal` variable evaluates to false, so the `request-only-once` and `request-withdrawal` rules are not applicable.

The `action-is-approve` variable evaluates to false, so the `approve-only-requested`, `approve-only-once`, `not-raised-by-approver` and `approve-withdrawal` rules are not applicable.

The `relevant-history` variable evaluates to a bag containing the two `entity` values from the `history` attribute with `transaction-id` equal to `61b9081d-92f1-46af-aa81-4f8454877619`, i.e., Carol's request and Bob's approval. Dave's request doesn't match the `transaction-id`.

The `make-withdrawal` rule evaluates to `Permit` because the `action-id` is `withdraw`, the subject is an accountant in the Finance Department and the `relevant-history` contains an ***action history record*** with `action-id` equal to `approve`. This rule contributes an `end-history` obligation to the result indicating that the ***action history records*** for this transaction can be discarded. The policy evaluates to `Permit` overall.

Upon receipt of the response the PEP discards the ***action history records*** for the indicated combination of resource (`http://example.com/account/payroll`), constraint (`withdrawal`) and transaction (`61b9081d-92f1-46af-aa81-4f8454877619`). Dave's action history record will be retained, at least until `2022-10-13T12:15:00Z`.

9 Architectural Considerations

[Informative]

This profile describes the ***SoD obligations*** being processed by PEPs, but this is not meant to preclude architectures where an intermediary system between the PEP and PDP processes the ***SoD obligations*** on behalf of a PEP.

The intermediary would store the ***action history records***. On receiving an authorization request from a PEP it would check its store of ***action history records*** for any relevant records, augment the authorization request with those records and forward the augmented request to the PDP. The intermediary would intercept the authorization response from the PDP, check for and process any ***SoD obligations***, convert the decision to "`Deny`" if necessary, strip the ***SoD obligations*** from the response and return the modified response to the PEP.

Such an intermediary would be useful for supporting ***SoD constraints*** in the presence of PEPs that do not implement the ***SoD obligations***, or PEPs that operate on the same resources but are unable to share the ***action history records*** for those resources.

It is feasible for the capabilities of an intermediary to be implemented in the context handler **[XACML-v3.0-Errata01-complete]**.

The intermediary won't be aware that a resource has been deleted, and therefore that the ***action history records*** relating to that resource can be discarded, unless it spots in passing an authorization request to delete the resource. It is not necessarily the case that deletion of the resource will entail a preceding authorization request or that the request will explicitly reference the resource. Policy writers should use ***transaction time limits*** to avoid the accumulation of inactive ***action history records*** in the intermediary.

10 Support for Policy Editing

[Informative]

Although the policies in the examples in this specification were designed from the point of view of illustrating key concepts and capabilities, they nonetheless show certain similarities due to common requirements for ***SoD constraints***. For example, both policies have rules to ensure that actions are performed in the correct order (`approve-only-raised`, `approve-only-requested`) and that no actions are repeated (`raise-only-once`, `approve-only-once` and `request-only-once`). Both policies have a primary rule that tests whether the subject is attempting an action that is in conflict with an action they performed earlier (`not-raised-by-approver`, `not-requested-by-approver`). The obligation expressions for generating `add-history` obligations are very regular, as is filtering the received ***action history records*** for relevance to the constraint and transaction. These commonalities point to the potential for much of the policy composition for an ***SoD constraint*** to be automated rather than requiring the policy writer to construct the policies from scratch. The main points of difference, that are highly variable and require the expertise of the policy writer, are the expressions that qualify who is eligible to perform each of the actions. These expressions are part of the otherwise predictable rules that permit the actions and produce the ***SoD obligations***.

A PAP specialized for supporting ***SoD constraints*** could present the policy writer with an interface that allows an ***SoD constraint*** to be described in simple, high-level terms. That is, a description of the actions, their required sequencing and the combinations that cannot be performed by the same person. The basic policy framework can then be automatically produced from this description leaving only the final details, such as who is eligible to perform each action, to be elaborated using a more generic XACML expression editing interface (or perhaps a simplified interface to cover the most common use cases). The bulk of the first example can be automatically constructed from knowing that there are two actions, `raise` and `approve`, that must be performed in that order by different users, that users are identified by email address in `subject-id` and that there is only one transaction per purchase order resource. The bulk of the second example can be automatically constructed from knowing that there are three actions, `request-withdrawal`, `approve` and `withdraw`, that must be performed in that order, though only the first two must be performed by different users, that users are identified by email address in `subject-id` and that there can be many transactions per account resource. It remains for the policy writer to define, via XACML expressions, who can perform each of the actions.

The sequence of actions making up a transaction comprise a workflow involving requestors and approvers. This workflow may be supported by the user interface of a bespoke application or it may be supported by a general-purpose workflow engine. In the latter case, there are opportunities for closer integration with a PAP. The workflow engine will have the means for a workflow designer to define the steps in a workflow and any restrictions or requirements that may apply to those steps. It may even explicitly call out ***SoD constraints***. The workflow description provides much of the information needed to produce XACML policies for ***SoD constraints***. If the workflow engine is able to export a machine-readable description of the workflow then a PAP could be extended to read that description, generate most of the policy framework and prompt the policy writer to fill in the final details. Even better would be a workflow engine that can generate the XACML policies itself (or a PAP that manages workflows) after offering the workflow designer the opportunity to define non-trivial conditions as XACML expressions. Essentially, XACML becomes the underlying enforcement mechanism for the workflow engine.

11 Conformance

An implementation claiming conformance with this specification MUST support the functions defined in Section 5 and the obligations defined in Section 7.

Appendix A. References

This appendix contains the normative and informative references that are used in this document.

While any hyperlinks included in this appendix were valid at the time of publication, OASIS cannot guarantee their long-term validity.

A.1 Normative References

The following documents are referenced in such a way that some or all of their content constitutes requirements of this document.

[RFC2119]

Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174]

Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[XACML-v3.0-Errata01-complete]

eXtensible Access Control Markup Language (XACML) Version 3.0 Plus Errata 01. Edited by Erik Rissanen. 12 July 2017. OASIS Standard incorporating Approved Errata. http://docs.oasis-open.org/xacml/3.0/errata01/os/xacml-3.0-core-spec-errata01-os-complete.html. Latest version: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-en.html.

[xacml-3.0-nested-ent-v1.0]

XACML v3.0 Related and Nested Entities Profile Version 1.0. Edited by Steven Legg. 16 February 2021. OASIS Committee Specification 02. https://docs.oasis-open.org/xacml/xacml-3.0-related-entities/v1.0/cs02/xacml-3.0-related-entities-v1.0-cs02.html. Latest stage: https://docs.oasis-open.org/xacml/xacml-3.0-related-entities/v1.0/xacml-3.0-related-entities-v1.0.html.

A.2 Informative References

The following referenced documents are not required for the application of this document but may assist the reader with regard to a particular subject area.

[RFC3552]

Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <https://www.rfc-editor.org/info/rfc3552>.

Appendix B. Security and Privacy Considerations

Enforcement of separation of duties constraints requires the accurate identification of the subjects who are performing the actions. A user could bypass the constraints by performing conflicting actions under separate user accounts with separate subject identifiers. Obviously, one way to mitigate this possibility is to apply administrative procedures to prevent a user obtaining multiple accounts, including access via shared accounts. Alternatively, if the multiple accounts that a user holds can be distinguished in some way then the policies could be written so that only accounts of a particular, tightly-administered kind are eligible to access resources protected by ***SoD*** constraints.

A user could also bypass the constraints if the subject identifiers change over time or have alternative values. If all the previous and alternative identifiers are kept with a user's records (e.g., in the PIP) and the context handler has access to these identifiers then the PDP can match ***action history records*** against all known identifiers for the user.

The examples in this specification use the `subject-id` attribute from the request to identify the user, but this is not a requirement. Any subject attribute could be used to fill the `subject-id` attribute in an ***action history record***. In particular, if users have a unique, immutable identifier such as an employee number then that should be used to identify users in ***action history records***. The policies should be written to deny access if the chosen attribute is not available for a subject. Multiple accounts for the same user are allowable if all the accounts for that user have the identical value for the immutable identifier.

Appendix C. Acknowledgments

C.1 Special Thanks

Substantial contributions to this document from the following individuals are gratefully acknowledged:

Hal Lockhart, Individual Member
Bill Parducci, Individual Member

C.2 Participants

The following individuals were members of this Technical Committee during the creation of this document and their contributions are gratefully acknowledged:

Steven Legg, ViewDS Identity Solutions
Hal Lockhart, Individual Member
Bill Parducci, Individual Member

Appendix D. Revision History

Revisions made since the initial stage of this numbered Version of this document may be tracked here.

Revision	Date	Editor	Changes Made
WD 01	17 August 2022	Steven Legg	Initial draft.
WD 02	12 October 2022	Steven Legg	Started adding examples.
WD 03	7 December 2022	Steven Legg	Added requests and results to the first example. Fixed a bug in the first example policy. Added the policy for the second example.
WD 04	17 February 2023	Steven Legg	Outside of an ***action history record*** the `transaction-id` attribute makes more sense as an action category attribute. This philosophical change required a small change to the policy in the second example. Added supporting text, requests and results to the second example.
WD 05	29 March 2023	Steven Legg	Added text on automating the generation of ***SoD constraint*** policies to the Support for Policy Editing section. Added text to the Security and Privacy Considerations appendix.
WD 06	24 May 2023	Steven Legg	Fixed missing hyphen in entity data-type URIs. Fixed missing `dateTime-one-and-only` function applied to `current-dateTime` attribute designators in second example policy. Fixed non-ASCII hyphens in dateTime values. Corrected the at-least-one-member-of function in the approve-purchase-order example rule. Fixed the action value in the final example request. Fixed XML indenting in examples 8.1.2 and 8.1.3. Added some text to Support for Policy Editing.

			Added some text to Security and Privacy Considerations.
WD 07	20 July 2023	Steven Legg	Relaxed the data-type restriction on the `transaction-id` attribute since the first example was successfully using anyURI. Added text to sections 7 and 9 on the wisdom of using the `time-limit` attribute to ensure the clean up of ***action history records***. Eliminated some non-breaking hyphens.

Appendix E. Notices

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website: [https://www.oasis-open.org/policies-guidelines/ipr/].

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OASIS AND ITS MEMBERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THIS DOCUMENT OR ANY PART THEREOF.

As stated in the OASIS IPR Policy, the following three paragraphs in brackets apply to OASIS Standards Final Deliverable documents (Committee Specifications, OASIS Standards, or Approved Errata).

[OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this deliverable.]

[OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this OASIS Standards Final Deliverable. OASIS may include such claims on its website, but disclaims any obligation to do so.]

[OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this OASIS Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Standards Final Deliverable, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.]

The name "OASIS" is a trademark of OASIS, the owner and developer of this document, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, documents, while reserving the right to enforce its marks against misleading uses. Please see https://www.oasis-open.org/policies-guidelines/trademark/ for above guidance.