UConML format specification

Author: Enrico Carniani
Issuer: CNR IIT - Pisa (IT)
Revision date: March, 17th, 2015 (b)

This document hosts the currently implemented elements and attributes as of Retrail version 2.0 for the UCon declaration file, which is a XML representation of its internal configuration. This file itself, along with the PIP implementations, solely defines the UCon behaviour for Usage Control and all the systemic parameters, such as timeouts or property values.
There it follows a quick guide, made step by step, specifying the semantics and usage of each element. All the UCon elements are defined in the ucon namespace, formally declared as **http://security.iit.cnr.it/retrail/ucon**.

Property

Tag Name: Property
Description: The root element to be declared on top of the configuration.
Location: <any element corresponding to an UCon component implementation>
Attributes: name: the name of the property to be set. Mandatory.
Mandatory: No.
Examples: <ucon:Property name=”timer”>2</ucon:Property>
<ucon:Property name="reputation(fedoraRole)">bronze</ucon:Property>

Each element defined in the file under the root level has its own live counterpart, implemented as a component of the UCon server architecture. Such component is treated as a sort of bean, which may hold some attributes externally accessible via setters and getters.
Provided that a component has such accessors, related properties get visible at UConML level as well, by specifying a Property element whose “name” attribute holds the property name. The content, be it textual, numeric, or even a structured node, is automatically converted and conveniently set by the UCon system when loading the configuration. This applies to all UCon components visible from the UConML, such as the root level itself, the PIP chain, any PIP, the behavioural automaton, and so on.
It is worth noting that even mapped properties are implemented. If a bean is backing up a property via a Map<> interface, it is possible to set its values by setting the name attribute with the syntax: “attributeName(keyToBeSet)”.

Config

Tag Name: Config
Description: The root element to be declared on top of the configuration.
Location: <xml root>
Mandatory: Yes.

Example: <ucon:Config xmlns:ucon=”http://security.iit.cnr.it/retrail/ucon”>...</ucon:Config>

This element is simply the root of all elements, and acts as a container for all the configuration.

Available properties

- **watchdogPeriod(double):** number of seconds for the periodic watchdog;
- **maxMissedHeartbeats(int):** number of heartbeats that PEPs are allowed to miss before the related sessions are declared stale (and thus, automatically cleaned up and removed by the UCon server).

PIPChain

Tag Name: PIPChain
Description: An optional container declaring a sequence of PIPs to be used as filters.
Location: Child of <Config>.
Mandatory: No, but its definition is strongly recommended.
Example: <ucon:PIPChain>...</ucon:PIPChain>

The container, if declared, substitutes the PIPChain loaded with the previous configuration. If not declared, the previous chain of filters is still held. Since this is a rule provided as a commodity but it is breaking the complete statefulness specification for the file, it is strongly recommended to specify the PIPChain in each configuration. It is allowed to have a completely empty chain.

PIP

Tag Name: PIP
Description: Declares and insert a PIP filter in the chain.
Location: Children of <PIPChain>.
Attributes: class: the canonical name of the filter class to be instanced. Mandatory.
Mandatory: No.
Example: <ucon:PIP class="it.cnr.iit.retrail.server.pip.impl.PIPSessions">...</ucon:PIP>

Creates a PIP and inserts it it the PIP chain. The filter is enabled to receive UCon events and modify the environment as it wishes. PIP will receive events in the cascading order they are declared. Note that the class attribute enables custom PIPs to be defined and used as well as the system provided ones: they simply have to adhere to the PIPInterface class.

Available properties

Even though any user property may be defined by the implementation, these are the standard ones implemented in the base PIP class that may be derived to obtain new custom PIPs:

- **uuid(string):** set the universal identifier. The canonical name of the class is set by default if this property is not set explicitly (this is quick but won't work with multiple instances of the same PIP in the declaring PIPChain);
- **issuer(string):** set the issuer uri for the possible attributes produced by the PIP. By default, the value is the UCon namespace uri.

Conventionally, if a custom PIP is able to produce a single attribute, it is recommended to adopt a property named "attributeId", in order to specify the attribute name that the pip is going to produce.

Behaviour

Tag Name: Behaviour
Description: Declares the behavioural automaton describing the usage control to perform.
Location: Child of <Config>.
Mandatory: Yes.
Example: <ucon:Behaviour>...</ucon:Behaviour>

Container that holds the definition of the behavioural automaton for usage control.

Available properties

- **lockTimeout(double):** number of seconds to wait for a busy session before giving up. It that case, a TimeoutException is thrown. It is important to specify such a value since in real production systems there's always something that could go wrong, for example a bad behaving PIP that is holding the lock too much time while some other clients would gain access to the resource.

States

Tag Name: States
Description: Declares a container for the states of the automaton.
Location: Child of <Behaviour>.
Mandatory: Yes.
Example: <ucon:States>...</ucon:States>
Defines the container for the State elements.

State

Tag Name: State
Description: Declares a state for the automaton.
Location: Children of <States>.
Attributes: name: the name of the state. Mandatory.
type: BEGIN (the initial state, mandatory),
END (one of the final states, at least one must be present),
ONGOING (perform continuous usage control here),
PASSIVE (wait for PEP explicit action)

It's mandatory.

Mandatory: Yes.

Example: <ucon:State name="INIT" type="BEGIN"/>

Defines the states for the automaton, one for each element declaration. The automaton must have exactly one BEGIN state, where a session starts from, and may have one or more final states, where the session is automatically terminated by the UCon (each DAL dependency is cleared out).

Zero or more ONGOING states may be present. In each ONGOING state, the related policy is automatically applied by the system in order to perform automatic continuous usage control over the sessions.

A PASSIVE state, whose presence in the automaton is not mandatory anyway, is instead a state where no check is done: the system simply waits for a PEP to do some other action to move on. For example, in the standard UCon model, the revocation state is PASSIVE, since the UCon would wait for the PEP to execute an endAccess action.

Actions

Tag Name: Actions

Description: Declares the actions container for the automaton.

Location: Children of <Behaviour>.

Mandatory: Yes.

Example: <ucon:Actions>...</ucon:Actions>

Defines the container for the Action elements. Each action works as a connector (an arch) from a source state to a target state, and completes the automaton diagram.

Action

Tag Name: Action

Description: Declares an action for the automaton, moving from one state to another.

Location: Children of <Behaviour>.

Attributes: class: the canonical name of the action class to be instanced. Mandatory.
source: the name of the source state the action is originating from. Mandatory.
target: the name of the target state the action is moving to, if no fail occurs.
May not be specified. In that case, target == source is assumed.

Mandatory: Yes.

Example:
```
<ucon:Action name="startAccess"
  class="it.cnr.iit.retrail.server.behaviour.PDPAction"
  source="TRY" target="ONGOING">...</ucon:Action>
```

The Action element creates an arch between the source and target states. The action is connected to a related policy, which discriminates if the action is to be taken or if it failed for some reason (Deny, Indeterminate, or NotApplicable decision made by the backing PDP component). Special destinations may be applied depending on the failing decision; these may be defined by the Target sub-element.

The policy to be applied is expressed in the XACML format as a direct child of this element as well. If not present, the system assumes a permanent Permit decision with no obligation is to be taken. This is a very convenient shortcut that also shorten the execution time, since no PDP evaluation is done at all.
It is worth noting that the class parameter allows for custom implementations to be done (they simply need to adhere the PDPAction interface). The basic implementation is it.cnr.iit.retrail.server.behaviour.PDPAction, which needs an additional “name” attribute to name the action (this is the name that the PEP must use to invoke this action).
There are 3 special builtin classes that may be also specified:

- it.cnr.iit.retrail.server.behaviour.TryAccess: used for a PEP standard tryAccess to open a session.
- it.cnr.iit.retrail.server.behaviour.OngoingAccess: used internally by the system; may be declared only from ONGOING source states.
- it.cnr.iit.retrail.server.behaviour.EndAccess: used for a PEP standard endAccess to terminate a session.

Target

Tag Name: Target
Description: Declares an additional failure target, based on PDP decision.
Location: Children of <Action>.
Attributes: decision: Deny, Indeterminate, or NotApplicable. Mandatory.
state: the name of the target state to go in case of that decision. Mandatory.
Mandatory: No. Any unspecified decision target is assumed not to move the automaton.
Example: <ucon:Target decision="Deny" state=”REJECTED”/>

A powerful yet easy way to determine where to move in case of a PDP decision fail. The Permit case is not allowed here, since it is covered by the target attribute of the Action element.

UConML sample

For the sake of clarity, here it follows a complete sample of a UConML configuration file.

```
<?xml version="1.0" encoding="UTF-8"?>
<ucon:Config xmlns:ucon="http://security.iit.cnr.it/retrail/ucon">
 <ucon:PIPChain>
  <ucon:PIP class="it.cnr.iit.retrail.server.pip.impl.PIPSessions">
       <ucon:Property name="uuid">sessions</ucon:Property>
  </ucon:PIP>
 <ucon:PIP class="it.cnr.iit.retrail.test.TestPIPReputation">
      <ucon:Property name="uuid">reputation</ucon:Property>
      <ucon:Property name="attributeId">reputation</ucon:Property>
      <ucon:Property name="reputation(user1)">bronze</ucon:Property>
      <ucon:Property name="reputation(user2)">bronze</ucon:Property>
      <ucon:Property name="reputation(user3)">bronze</ucon:Property>
      <ucon:Property name="reputation(user4)">none</ucon:Property>
```

```
  </ucon:PIP>
  <ucon:PIP class="it.cnr.iit.retrail.server.pip.impl.PIPTimer">
       <ucon:Property name="uuid">timer</ucon:Property>
       <ucon:Property name="attributeId">timer</ucon:Property>
       <ucon:Property name="resolution">0.25</ucon:Property>
       <ucon:Property name="forStateType">ONGOING</ucon:Property>
   </ucon:PIP>
 </ucon:PIPChain>
 <ucon:Behaviour>
   <ucon:States>
        <ucon:State name="INIT" type="BEGIN"/>
       <ucon:State name="TRY" type="PASSIVE"/>
       <ucon:State name="ONGOING" type="ONGOING"/>
       <ucon:State name="REVOKED" type="PASSIVE"/>
       <ucon:State name="REJECTED" type="END"/>
       <ucon:State name="DELETED" type="END"/>
    </ucon:States>
    <ucon:Actions>
        <ucon:Action class="it.cnr.iit.retrail.server.behaviour.TryAccess" source="INIT"
target="TRY" />
        <ucon:Action name="startAccess" class="it.cnr.iit.retrail.server.behaviour.PDPAction"
source="TRY" target="ONGOING">
        <Policy xmlns="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17"
PolicyId="tryStart-policy"

RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:first-applicable"
Version="1.0">
                      <Description>Test on-authorization policy</Description>
                      <Target></Target>
                      <Rule Effect="Permit" RuleId="test:rule">
                      <Target>
                             <AnyOf>
                             <AllOf>
                                    <Match
MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
                                    <AttributeValue
DataType="http://www.w3.org/2001/XMLSchema#string">bronze</AttributeValue>
                                    <AttributeDesignator AttributeId="reputation"

Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject"
                                           DataType="http://www.w3.org/2001/XMLSchema#string"
MustBePresent="true" Issuer="http://security.iit.cnr.it/retrail/ucon">
                                    </AttributeDesignator>
                                    </Match>
                                    <Match
MatchId="urn:oasis:names:tc:xacml:1.0:function:integer-greater-than">
                                    <AttributeDesignator AttributeId="openSessions"

Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject"

DataType="http://www.w3.org/2001/XMLSchema#integer" MustBePresent="true"
Issuer="http://security.iit.cnr.it/retrail/ucon">
                                    </AttributeDesignator>
                                    <AttributeValue
DataType="http://www.w3.org/2001/XMLSchema#integer">2</AttributeValue>
                                    </Match>
```

```
                        </AllOf>
                        </AnyOf>
                </Target>
                </Rule>
                <Rule Effect="Deny" RuleId="test:rule2"/>
        </Policy>

        </ucon:Action>
        <ucon:Action class="it.cnr.iit.retrail.server.behaviour.EndAccess" source="TRY"
target="DELETED"/>
        <ucon:Action class="it.cnr.iit.retrail.server.behaviour.OngoingAccess"
source="ONGOING">
        <ucon:Target decision="Deny" state="REVOKED" />
        <ucon:Target decision="NotApplicable" state="REVOKED" />
        <ucon:Target decision="Indeterminate" state="REVOKED" />
        <Policy xmlns="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17"
PolicyId="on-policy"

RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:first-applicable"
Version="1.0">
                <Description>Test on-authorization policy</Description>
                <Target></Target>
                <Rule Effect="Permit" RuleId="test:rule">
                <Target>
                        <AnyOf>
                        <AllOf>
                                <Match
MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
                                <AttributeValue
DataType="http://www.w3.org/2001/XMLSchema#string">bronze</AttributeValue>
                                <AttributeDesignator AttributeId="reputation"

Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject"
                                        DataType="http://www.w3.org/2001/XMLSchema#string"
MustBePresent="true" Issuer="http://security.iit.cnr.it/retrail/ucon">
                                </AttributeDesignator>
                                </Match>
                                <Match
MatchId="urn:oasis:names:tc:xacml:1.0:function:double-greater-than">
                                <AttributeDesignator AttributeId="timer"

Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject"
                                        DataType="http://www.w3.org/2001/XMLSchema#double"
MustBePresent="true" Issuer="http://security.iit.cnr.it/retrail/ucon">
                                </AttributeDesignator>
                                <AttributeValue
DataType="http://www.w3.org/2001/XMLSchema#double">3</AttributeValue>
                                </Match>


                        </AllOf>
                        </AnyOf>
                </Target>
                </Rule>
                <Rule Effect="Deny" RuleId="test:rule2"/>
        </Policy>
```

```
            </ucon:Action>
            <ucon:Action class="it.cnr.iit.retrail.server.behaviour.EndAccess"
source="ONGOING" target="DELETED"/>
            <ucon:Action class="it.cnr.iit.retrail.server.behaviour.EndAccess"
source="REVOKED" target="DELETED"/>
       </ucon:Actions>
 </ucon:Behaviour>
</ucon:Config>
```