A Use-based Approach for Enhancing UCON

Christos Grompanopoulos, Antonios Gouglidis and Ioannis Mavridis

Department of Applied Informatics, University of Macedonia,
Egnatia Street 156, 54006 Thessaloniki, Greece
{groban,agougl,mavridis}@uom.gr

Abstract. The security related characteristics of entities, the contextual information that describes them and the previous or concurrent usages exercised in the system are the criteria that the Usage CONtrol (UCON) family of models utilizes in the usage decision process. In this paper, a detailed classification of the aforementioned criteria along with a representative usage scenario for each category is presented, unveiling a number of UCON's limitations. In turn, a Use-based Usage CONtrol (UseCON) model is proposed that provides, for the creation of a usage decision, enhanced handling of information regarding context and previous or current usages exercised in the system. The enhanced capabilities of the proposed approach are demonstrated and discussed with the use of detailed application examples.

Keywords: Access control, usage control, UCON, UseCON

1 Introduction

Controlling the access to the resources of a system is an essential requirement for every computer security system [2]. Traditional access control models utilize only a single criterion for the allowance of an access request, which is related to the security characteristics of the subject and object involved in the requested access [7]. More specifically, whenever a subject requests to access an object, the subject's clearance and the object's classification in Mandatory Access Control (MAC) models [8], the subject's identity in Discretionary Access Control (DAC) models [6] and an activated role from a set of authorized to the subject in Role Based Access Control (RBAC) models [9], are being utilized accordingly. Attribute based access control approaches [4] provide enhanced flexibility, when compared to the aforementioned access control models, by utilizing a number of subject and object security related characteristics, which are expressed in the form of attributes.

The Usage CONtrol (UCON) family of models [5] provides an integration of traditional access control, digital rights and trust management. Moreover, UCON encompasses attribute-based characteristics, along with the concepts of continuity of decision and attribute mutability. Through the utilization of continuity of decision in UCON, access control to a resource is being controlled either continuously through an ongoing rule, or only before an access is permitted through a pre rule, as in traditional access control models. Therefore the term usage is preferred to be used instead of access. Moreover, the complexity of modern computing environments requires the utilization of a number of criteria during the usage control decision making process. UCON, employs three criteria for the creation of a usage decision, namely, the security related characteristics (henceforth called *properties*), contextual information and information regarding previous or current usages of the system's entities. However, whenever a subject s requests the usage of an object o, the usage control decision making can be based on either information related to s and/or o, or on information related to other system entities (e.g. father's *properties* may have an influence on the son's permissions), and henceforth mentioned as *direct* and *indirect* entities of the requested usage, respectively.

Despite the fact that the usage control decision making process in UCON utilizes all the three criteria, these are commonly related only to the *direct* entities. Additionally, the attribute mutability mechanism of UCON introduces a number of limitations regarding the utilization of information about previous/current system usages. For example, no information about previous requested usages that were denied is recorded and no discrimination is done between the usages that have been revoked by the usage control system and the usages that have been terminated by a subject's request. Consequently, attribute mutability is unable to support a policy rule that is based on historical information regarding revoked usages. Moreover, modern computing environments present novel and complicated usage modes performed on objects by subjects, which are poorly supported through right entities in UCON. These complex operation modes require additional information that is essential for their execution, unlike the simple and straightforward operation modes that were previously supported by traditional access control models, e.g. read, write and execute operations in an operating system. For example, a banking transaction encompasses additional information, which is necessary for its operation, like the amount of transfer, the execution date etc.

This paper continues with a detailed categorization of the usage decision criteria utilized in UCON along with representative usage scenarios. Additionally, Sect. 2 highlights the challenging issues of utilizing the usage decision criteria in UCON. Section 3 proposes a new usage control model that extends UCON in order to provide mainly an enhanced utilization of the usage decision criteria and support for complicated usage modes. Application examples of the enhanced capabilities of the proposed model are presented in Sect. 4, and our conclusions are given in Sect. 5.

2 Utilization of Decision Criteria in UCON

UCON is a next generation access control model capable of evaluating a number of usage decision criteria for the allowance or not of a usage request. Nevertheless, a limited utilization of the aforementioned criteria, related to the *indirect* entities, is being noticed. A detailed description of the criteria's utilization, along with corresponding representative usage scenarios[1], follows.

2.1 Security Characteristics of Entities

Security characteristics of system entities in UCON, are associated with subject and object attributes. These attributes are utilized by functional predicates (authorizations) that are evaluated for the usage decision. An example of a usage scenario, where only the subject's and object's *properties* are taken into consideration during the usage decision making process, is implementation of a MAC policy in UCON as presented in [5]. More specifically, in a system that implements a MAC policy the following rules apply:

Usage Scenario 1. *A clearance attribute is assigned to all the subjects of the system. Moreover, a classification attribute that shares the same value domain with clearance, is also assigned to all the objects of the system. A relation exists between the values of clearance and classification, thus creating a form of hierarchy. Consequently, a subject can read an object only if its clearance overcomes the object's classification. In addition, an object can be written by a subject only if its classification overcomes a subject's clearance.*

Implementing the usage scenario 1 in UCON requires the utilization of authorization predicates that are evaluated on subject and object attributes. It is worth mentioning that UCON utilizes attributes for two purposes. More specifically, UCON does not only associates the entity's *properties* into the attribute values but also records into them the execution of system usages through the attribute mutability mechanism. Nevertheless, the values of the attributes that associate the *properties* of the entities are not updated automatically by the usage control system (update procedures) but only after the intervention of an administrator.

A limitation in UCON's authorizations is the fact that only the attributes from the *direct* entities are utilized. Nevertheless, in modern access control scenarios, it is possible that *properties* from *indirect* entities could also affect the authorization evaluation. A representative usage scenario that falls into the latter category is the following:

Usage Scenario 2. *Bob is a subscriber to a golf club that provides an amusement park for the children of its members. Bob's daughter, Alice is permitted to use all the available toys except from the carousel. Alice is permitted to use the carousel only if her father (Bob) is a member of the golden club category.*

When attempting to support the usage scenario 2 in UCON, Alice and carousel are considered to be the *direct* entities of the requested usage. However, during the usage decision making process, the values of Bob's attributes

[1] All the usage scenarios presented in this paper refer to pre authorization policy rules. However, the same criteria can also be applied to ongoing authorization rules.

(e.g. his golden category membership) are also required. Due to the fact that Bob is an *indirect* entity, his attributes are not directly utilized in the corresponding authorization predicate. Nevertheless, if Alice is supported by an attribute "father" (that is assigned with the value of "Bob") then UCON is capable of resolving Bob's attribute values and consequently utilize them for the usage decision.

2.2 Contextual Information

Contextual information in UCON is associated with special system variables, which are entitled condition variables. These variables are utilized in condition predicates in order to create a usage decision. A usage scenario, as originally presented in [5], that requires the utilization of contextual information for the creation of the usage decision follows:

Usage Scenario 3. *The members of an institution are categorized into "faculty" and "student". The same categorization is also applied to the institution's areas. A member of a specific category (e.g. faculty) can exercise a right only in areas having the corresponding label (e.g. faculty areas).*

The presented approach in [5], proposes the evaluation of condition predicates that contain condition variables, which associate the location information with *direct* entities. However, in case where contextual information that is associated with the *indirect* entities is required for the usage decision, UCON seems to be incapable of resolving which condition variable represents the contextual information that is related with a particular system entity. A usage scenario where contextual information, which is associated with the *indirect* entities, is utilized for the usage decision follows:

Usage Scenario 4. *The doctors in a hospital are categorized into "seniors" and "juniors" in respect of their operational experience. Every "junior" doctor is supervised by a corresponding "senior" doctor. Whenever a "junior" doctor, named Alice, sets a request for the execution of a specialized operation, e.g. an open heart surgery, a policy directive requires the physical coexistence of Alice's "senior" doctor supervisor, named Bob.*

A policy rule in UCON that models the usage scenario 4 requires the comparison between two locations represented by two separated condition variables. The problem arises from the fact that Bob is an *indirect* entity. In such a case, specifying in UCON the particular condition variable that represents Bob's location seems to be impossible. In usage scenario 4, Alice is the *direct* entity of the usage request and only information related with her is utilized for the usage decision (condition). Even if UCON can represent with an Alice's attribute the fact that Bob is her supervisor, it is not possible to link at the same time Bob with a condition variable that represents his location. A solution could be provided by utilizing a number of condition variables that represent contextual information, which are irrelevant with the *direct* entities of the usage (e.g. "subjectsSupervisorLocation" may be the condition variable that represents the location of Bob). However, in a system with a large number of condition variables, such an implementation could result in a very complicated usage control system.

2.3 Historical Information of Usages

There are cases where historical information of previous or current usages executed by the *direct* entities, are needed to be utilized for usage decision. A usage scenario, where the previous usages of a subject may affect the allowance of a new usage requested by the same subject, is the following:

Usage Scenario 5. *An on-line collaborative educational software provides to its members the capability to post questions that can be answered by other members. However, a policy rule requires that a member is allowed to set a new question only if he had previously provided at least two answers to questions of other members.*

UCON is capable to support usage scenario 5 either through authorizations that incorporate attribute update procedures or through obligations. Specifically, if answering a question is considered to be a system usage, then each time a member provides an answer, the value of an attribute that records this usage is being updated (attribute mutability). Consequently, an authorization predicate is evaluated, based on the value of the aforementioned attribute, to allow or not to the posting of a new question. More specifically, attribute mutability in UCON actually implements a mechanism that records the allowed system usages in attributes of *direct* entities. For example, every time a user listens to a music file, a specific attribute of her is being updated. The values of these attributes do not represent security characteristics of entities and are updated automatically by the attribute mutability mechanism. As a result, information about allowed usages is utilized for usage decision. However, attribute mutability faces a number of issues. Firstly, it provides limited knowledge regarding the system usages (only the allowed ones that contain attribute updates). Secondly, attribute mutability complicates the policy administration process by adding attribute update procedures to policy rules.

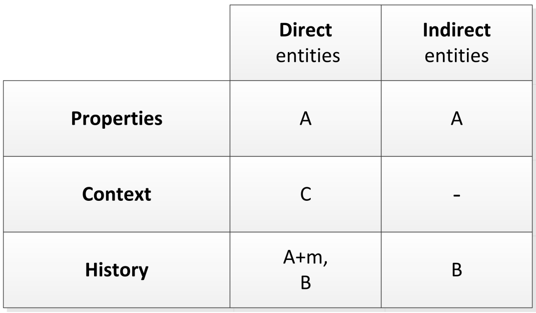
Giving an answer to the question in usage scenario 5, can be considered as an obligation operation that must be executed twice as a criterion for allowing a usage request to post a new question. Obligation operations in UCON also represent usages that are exercised by subjects on objects. However, these obligation operations are discriminated from normal system usages because they are not controlled by a decision factor (Authorization, oBligation or Condition) and can be performed whenever required [10]. Nevertheless, in modern computing environments, it is possible for the usage decision to be dependent on past usages of *indirect* entities. A usage scenario that falls into this category is the following:

Usage Scenario 6. *In a research institute, a presentation room is equipped with both an interactive board and a media player. A policy rule requires that* *an employee is permitted to access the media player only if there is no other presentation in progress (usage of the interactive board) in the same room.*

Usage scenario 6 can be modeled only through UCON's obligations and not through authorizations that incorporate attribute mutability update procedures (as happened with usage scenario 5). Authorizations with attribute mutability fail to model scenario 6 because only the attributes of the *direct* entities of a usage are being updated. Moreover, authorizations utilize only attribute values from *direct* entities. Thus, the usage of the media player in usage scenario 6 without the utilization of obligations, seems to be impossible. However, a significant drawback of obligations is the lack of a feasible fulfillment mechanism, as it is mentioned in [3].

Therefore, we summarize the utilization of UCONs usage decision criteria in Table 1, based on the analysis performed in the aforementioned scenarios. The usage decision criteria are represented as rows on the left side of the table. These are, as identified, the *properties*, *context* (contextual information), and *history* (information regarding previous or concurrent usages) of the system entities. The far right two columns of the table represent the origin of the aforementioned criteria, which can stem from either a *direct* or an *indirect* entity. Thus, each usage decision criterion, originating from an entity, is utilized by UCONs decision factors that are expressed in the corresponding cell. Each UCON decision factor is represented by a letter (Authorization, oBligation, Condition) combined, if required, with the attribute mutability mechanism. For instance, if a usage decision criterion is based on historical information stemmed from *direct* entities, then UCON is capable of utilizing it by using either authorizations with attribute mutability (A+m) or obligations (B).

Table 1. Utilization of decision criteria in UCON



	Direct entities	**Indirect** entities
Properties	A	A
Context	C	-
History	A+m, B	B

3 The proposed UseCON model

The UCON family of models [5] is mainly characterized by fine grained control of resources, support for continuity of decision, and attribute mutability. However, as it is highlighted in the previous section, UCON presents a number of limitations regarding the utilization of criteria originating from *indirect* entities. In the rest of this section, the proposed Use-based usage CONtrol (UseCON) model is presented, as an approach to overcome the previously mentioned limitations in modern computing environments.

3.1 Elements

The UseCON model consists of three elements viz. entities, attributes and authorizations. An entity is associated with attributes and authorizations are utilized as usage decision factors.

Entities. We define the set of *entities* (E) containing all the entities $e_i, i = 1, 2, \ldots, n$, of a system, in the form of subjects (s), objects (o), actions (a) and uses (u). Subjects are entities that request to exercise operations on objects. A subject can be a human, a device or a software agent acting on behalf of a human. Objects can be physical entities, logical entities or services (e.g. a printer, a file or a database migration service). An entity operating as a subject in one usage may be the object in another usage [11]. Actions are entities that represent the operations that subjects can exercise on objects. The types of subjects or objects determine the types of the actions that can be exercised on them. For example, in case of a file, a list of possible actions could be read, write and execute.

A core entity of the UseCON model is the *use* entity. A use materializes all the characteristics of a usage that are critical for the decision making process. A use actually records the relation between the subject, object and action of a particular usage. The information contained in a use is not predetermined but is composed at the time of a usage request. The use entity that materializes the usage under consideration is the *direct* use while all the others are *indirect* uses.

Attributes. Subjects and objects are associated with security-relevant characteristics and capabilities, called attributes. In addition, contextual information, which in UCON is stored in condition variables [5], is associated in UseCON with subject or object attributes. In order to support complicated operations in modern computing systems, it is required for actions to be associated with attributes, too. An example of an action attribute in a file-related operation (e.g. write), could be an encryption key. Uses also have attributes in order to encompass information that is related to a combination of subject, object and action (e.g. the price of a service).

The set of entity attributes (EA) contains the attributes $ea_i, i = 1, 2, \ldots, m$ of all entities. A relation ATT(e_i) denotes the association of an entity $e_i \in E$ with a tuple of attributes. We adopt the function notation in order to represent the value (range) that is assigned to an attribute (function) of a specific entity (domain). For example, in the expression Age(Alice) = 34, 'Alice' is an entity that has been associated with an attribute *Age* having a value of '34'. Every subject, object and action is associated with an *id* attribute, which has a unique value that remains constant during the life cycle of the usage control system [10]. When an instance of a use is created, it is associated with a tuple of attributes <*sid*, *oid*, *aid*> that have the same values with the particular *id* attributes of the *direct* entities (s, o, a) of the usage materialized by the use. Moreover, an additional *time* attribute is associated with each use [2]. The tuple <*sid*, *oid*, *aid*, *time*> is unique for each use (the usage of a subject on an object with an action at a specific time is also unique) and consequently operates as the identifier of the use.

Each use is further associated with a *state* attribute, which embodies the accomplished status of the usage in progress, as it is described in [10] and augmented in [1]. The *state* attribute represents the current state of a usage, as depicted in Fig.1, and each time it receives one of the following values:

- Requested: On a request for a usage, appropriate attributes are associated with the use and proper values are assigned to them. The pre-authorization rules, which govern the requested usage, have not been evaluated yet.
- Activated: The requested usage has been allowed, as a result of successfully fulfilled pre-authorization rules, and is being executed.
- Denied: The requested usage has been denied, because it failed to satisfy the pre-authorization rules.
- Stopped: The allowed / ongoing usage has been terminated by the system due to a violation of an ongoing authorization rule.
- Completed: The usage that has been completed due to a subject's intervention.

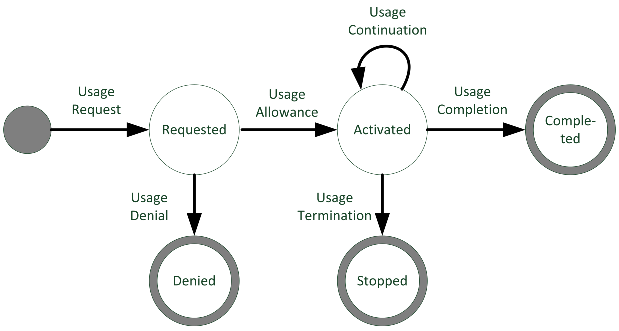


Fig. 1. Use state-transition diagram

Authorizations. UseCON's usage decision factor is authorizations. Authorizations are able to utilize all the three criteria described in the previous section,

[2] The value of the time attribute could vary from the time of usage request to the time of usage termination/completion and is left open as an implementation choice.

regardless they are originating from *direct* or *indirect* entities. A detailed description of authorizations follows:

- **Attribute dependent Authorizations:** AdAs are functional predicates that are evaluated on entity attributes. However, attribute values in UseCON contain both *properties* and contextual information of entities, excluding historical information of usages (as introduced by the attribute mutability mechanism in UCON).
- **Usage dependent Authorizations:** In UseCON, all the system usages are recorded with the help of the use entity. UdAs are functional predicates that are evaluated on historical information of usages. Thus, a requested usage can be permitted only if another usage has been previously exercised. For example, an UdA can model a policy rule requiring that a student can present the work of his team if and only if he has previously been registered in the system. However, UseCON's UdAs are able to support more complicated rules. Hardening the previous example, it may be required that the presentation of a team's work by the student is allowed if any member of his team has already been registered. UdAs are more flexible to utilize historical information of usages compared to UCON's authorizations combined with attribute mutability, due to the fact that they support historical information from both *direct* and *indirect* entities.

Associating contextual information with entity attributes, results in the replacement of UCON's conditions with authorizations. Moreover, operations required by UCON's obligations are handled as usages in UseCON. Therefore, exercising obligation operations in UseCON is verified by searching the history of *indirect* use. It is worth mentioning that post obligations, which are operations that must be fulfilled after the termination of a usage, are not supported by the proposed model and they are considered to be an administration issue. The UseCON elements and their relations are depicted in Fig.2.

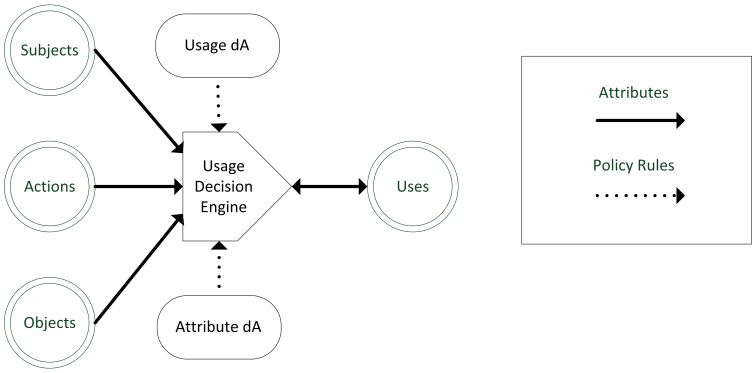


Fig. 2. UseCON usage control system

3.2 UseCON sub-models

UseCON utilizes only AdAs and UdAs as factors for usage decision. Applying continuity of decision results in four UseCON sub-models, as follows.

Pre-Attribute dependent Authorizations (preAdA). A subject is permitted to exercise an action on an object if a predicate preAdA is satisfied. There is no further (ongoing) control after the usage's allowance. More specifically, a preAda rule is defined as predicate that utilizes the attributes of the use that materializes the requested usage as follows [3]:

$$allowed(s,o,a) \Rightarrow preAdA\,(ATT(u)) \tag{1}$$

Pre-Usage dependent Authorizations (preUdA). A preUdA rule utilizes historical information of usages. More specifically, a preUdA rule is defined as a predicate that utilizes the attributes from both the *direct* use (u) and the *indirect* uses (u'). The number of the *indirect* uses that fulfill the preUdA predicate should satisfy a relational condition (less, greater or equal) with a specified natural number. The semantics of a preUdA rule is as follows:

$$allowed(s,o,a) \Rightarrow |\{u' \in U \colon preUdA\,(ATT(u), ATT(u'))\}| \otimes k \tag{2}$$

In the previous notation, $k \in \mathbb{N}$, the symbol $\otimes$ is replaced by a relational operator, U is the set of use entities and $|B|$ denotes the number of elements of set B.

Ongoing-Attribute dependent Authorizations (onAdA). An onAdA rule utilizes the same elements with preAdA and is defined as follows:

$$\begin{aligned} allowed(s,o,a) &\Rightarrow true \\ stopped(s,o,a) &\Leftarrow \neg onAdA(ATT(u)) \end{aligned} \tag{3}$$

The semantics of an onAdA functional predicate are the same with a preAdA one.

Ongoing-Usage dependent Authorizations (onUdA). An onUdA rule utilizes the same elements with preUdA and is defined as follows:

[3] A predicate is also able to utilize the attributes of the direct entities, by applying the reverse id^{-1} function (e.g. $id^{-1}(sid(u)) = s$) on the id values of the entities a. Moreover, if the value of an attribute from a direct entity is the id of an indirect entity then a predicate can also utilize its attribute values for the evaluation. The same applies to all UseCON's sub-models.

$$\begin{aligned} allowed(s,o,a) &\Rightarrow true \\ stopped(s,o,a) \Leftarrow |\{u' \in U \colon onUdA(ATT(u), ATT(u'))\}| &\otimes k \end{aligned} \tag{4}$$

The semantics of an onUdA functional predicate are the same with a preUdA one.

4 Examples of UseCON Enhanced Capabilities

UseCON model enhances UCON's fundamental design guidelines as continuity of decision and attribute based usage control by introducing a number of innovative modeling decisions. Specifically, UseCON directly associates entities with contextual information and also replaces UCON's rights with actions enhanced with attributes. The aforementioned decisions in combination with the augmented utilization of historical information through the support of the new *use* entity, results in enhanced capabilities, as demonstrated in the following examples.

4.1 Abstraction of actions

In UCON, rights correspond to permissions for subjects to execute usage functions on objects. However, rights are not described with attributes. The replacement of UCON's simple rights with UseCON's actions described by attributes, provides enhanced capabilities, as follows.

Simplifying the administration of policy rules. A UCON policy rule governs the allowance either for a specific right or all rights. Thus, every time a new right is introduced in the security system, the policy administrator should most likely create a corresponding policy rule that permits its usage. However, in a computing environment that encompasses a great number of rights, policy administration is becoming a complicated process. In UseCON, the description of actions by attributes provides the policy administrator with the capability to govern the allowance of a set of actions by a single policy rule, as presented in the following example.

Example 1. *A company that offers location discovery services provides the capability to its customers to require the location of an object. A customer, according to his classification, can request the location of an object with a desired accuracy level. For example, members of the "golden" category might request the location of an object with an accuracy expressed in meters, while regular users are able to request the location of an object in kilometers.*

Modeling example 1 in UCON requires the creation of a unique right entity for each accuracy level of the location discovery service. Moreover, the policy administrator must create an additional policy rule (authorization, condition or obligation) that governs the allowance of the particular right's request. Hence, it is impossible with UCON modeling to create a policy rule that governs the allowance of a subset of rights e.g. rights that model location discovery services.

However, the replacement of UCON rights with UseCON actions associated with attributes provides the capability to model the relation that possibly exists between actions. More specifically, in example 1, every action is associated with an attribute, named *type*. Actions that refer to location discovery services have a unique *type* attribute value e.g. "LocService". Thus, by utilizing the value of *type*, a policy rule is able to govern the allowance of all the actions that represent location discovery services.

The UseCON modeling of example 1 results into the following preAdA rules:

accuracy : $A \to W$	Location accuracy level supported by the service
category : $S \to C$	Customer's category. "Premium" or "Regular"
type : $A \to T$	Type of service. "LocService" for location discovery

$$\begin{aligned} allowed(s,o,a) \Rightarrow type(a) = \text{ ``LocService'' } \wedge category(s) = \text{ ``premium''} \\ allowed(s,o,a) \Rightarrow type(a) = \text{ ``LocService'' } \wedge category(s) = \text{ ``regular''} \\ \wedge\, accuracy(s) = \text{ ``kilometers''} \end{aligned}$$

The first preAdA rule governs the allowance of two actions (location discovery service with accuracy level of kilometers and location discovery service with accuracy level of meters). The additional accuracy attribute utilized in the second preAdA rule represents the accuracy level of the location discovery service e.g. "kilometers" or "meters".

Negotiating action parameters. The use of action's attributes in UseCON does not only simplify the policy administration process, as mentioned previously, but also provides enhanced capabilities for negotiating the action parameters of a usage request.

In UCON, a subject is able to request the usage of a specific right but it is not possible to request a "generic" right, e.g. the location of an object without specifying particular accuracy requirements. However, the utilization of the attribute *type*, as introduced in UseCON modeling of example 1, is further able to provide to subjects the capability to request the execution of a usage by only specifying the type of the action. Therefore, the subject of example 1 may request the execution of any action that contains the value "LocService" in the *type* attribute. When the UseCON decision creation engine receives such a request, it evaluates the policy rules that govern the allowance of actions with the specific value in the attribute *type*. Consequently, the usage control system does not respond with a simple allow or deny message, but with a list containing all the suggested actions that the subject is permitted to exercise. Thus, if the returned list is not empty, the subject can select the action that satisfies her needs and send a new request. The sequence of messages exchanged between the subject and the UseCON usage decision engine is depicted in Fig. 3.

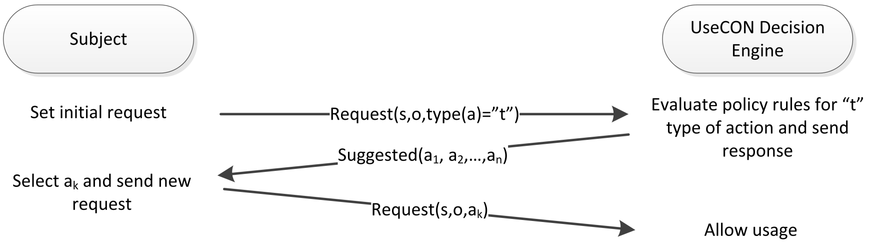


Fig. 3. Sequence of messages for action negotiation

Supporting Action Hierarchies. Relevant actions can participate in an action hierarchy. An example of action hierarchy in a hospital sector is presented in [5] where an action "a doctor writes a remedy on a patients record" is considered to be senior to the action "a doctor simply reads the patients medical history". The hierarchy of actions depends on the security policy of the particular usage control system. However, the policy rule for a senior action dominates on the policy rules for all its junior actions. A more detailed example is the following:

Example 2. *The security policy of a hospital defines that only doctors can read the medical history of a patient. However, altering a patient medical record is permitted only to doctors that have the same specialty with the category of the patient's illness.*

As UCON rights are not described with attributes, it seems impossible to model the relations between them and form a hierarchy. In UseCON, however, the classification of actions is possible through the utilization of action attributes. Consequently, both the policy administrator and the usage control mechanism are able to utilize such hierarchy information in order to enhance the expressiveness of the policy rules and to simplify the usage decision creation process, respectively. For example, whenever a subject requests the usage of two directly related actions, a proper usage control mechanism should evaluate only the policy rule that permits the senior action in the action hierarchy. In addition, in UseCON modeling of example 2, the policy administrator is capable of creating a rule that permits the execution of a read action on a medical record of a patient (a junior action), by examining if the requesting subject has previously exercised a write action on the medical record of any patient (a senior action). The modeling of example 2 with the use of a preUdA rule follows:

snr $: A \to 2^A$ The set that contains the ids of the senior actions

$$\begin{aligned} allowed(s,o,a) \Rightarrow \mid \{u' \in U\colon status(u') = \text{ ``completed'' } \wedge sid(u') = id(s) \wedge \\ oid(u') = id(o) \wedge aid(u') \in snr(a)\ \} \mid \geq 1 \end{aligned}$$

4.2 Utilization of usage information

The introduction of the *use* entity in UseCON provides new capabilities to the policy administrator. The utilization of use entities along with their attributes values provides the capability for enhanced utilization of historical information of usages and proper association of information to the system entities, as it is presented in the following examples.

Supporting Transactions. Some *properties* are not related with a single entity (subject or object), but with a combination of them. For example, an object attribute in UCON is associating information originating either directly from the object or from the right - object combination e.g. the price of the service [5]. Thus, if different rights can be exercised on an object, a separate *price* attribute for every one of these rights should be created. In addition, a detailed analysis unveils that the price of a service is actually associating information originating form the subject - object - right combination. More specifically, different customers may be charged with different prices for the execution of the same right on the same object. Therefore, the association of *properties* in a usage control system either with a single entity or with a usage is proposed. The former kind of information is associated with the related entity attributes while the latter with the corresponding use attributes.

While the values of entity attributes are set by an administrative operation, the creation of use entities and their corresponding attribute values are not predetermined but they are accomplished during the operation of the usage control system. More specifically, a subject entity and its attribute values are determined before the execution of any usage. However, a use entity and its attribute values are created only when a subject requests the corresponding usage. The values of use attributes should be assigned with rules that are application dependent and utilize the attribute values of the other entities participating in the usage. An example of information that is associated with use attributes is related to transactions. A transaction is a complicated system process that is composed from a set of particular system usages. In the UseCON model, every usage is modeled through a use entity that is associated with a *transaction* attribute. Uses that belong to the same *transaction* can share the same value of the *transaction* attribute. By utilizing proper values of use attributes, the policy administrator is able to define usage control rules with enhanced expressiveness. An example of the transaction attribute utilization in the creation of the usage decision follows.

Example 3. *In an accounting office the whole set of usages that update the files of a specific customer are forming a transaction. All these usages can be performed by a number of different employees and may concern a number of different files. However, because all these usages belong to the same transaction, they should be covered with the same privacy statement executed once by a single employee.*

In the following preUdA rule that models example 3 in UseCON, the execution of a *consent* action by any usage of the transaction is examined:

tr $: U \rightarrow T$ The name of the transaction where the usage belongs to

$$allowed(s,o,a) \Rightarrow \mid \{u' \in U \colon status(u') = \text{ “completed” } \wedge tr(u') = tr(u) \wedge aid(u') = \text{“consent” } \} \mid \geq 1$$

Enhanced utilization of historical usage information. Attribute mutability in UCON presents a number of limitations. For example, an attribute update procedure is executed only after the allowance of a requested usage. Thus, the denied usage requests are not recorded and information regarding such facts is not utilized for subsequent usage decisions. In addition, in an UCON ongoing rule, the same attribute update procedures will be executed if either the usage has been terminated by the subject or revoked by the usage control system, due to the ongoing rule violation. Consequently, UCON is incapable to discriminate the usages terminated by the subject from those revoked by the usage control system.

The UseCON model provides with comprehensive knowledge about the previous system usages through the utilization of the use entity. More specifically, the *state* attribute of a use entity provides the ability to discriminate between requested, active, denied, revoked and terminated usages. Such information can be utilized for future usage decisions. An example, where information about previously revoked usages is used for the creation of the usage control decision follows.

Example 4. *In a Digital Rights Management (DRM) system there is an upper bound limit on the number of simultaneous usages of an object by subjects. Whenever the maximum number of usages of an object is exceeded, several revocation strategies can be applied [5]. However, as a mean of policy fairness, the execution of a usage that has been previously revoked by the system is freely permitted without the evaluation of additional policy rules.*

The corresponding preUdA rule that implements the policy described in example 4 follows:

$$allowed(s,o,a) \Rightarrow \mid \{u' \in U \colon status(u') = \text{ “revoked” } \wedge sid(u') = sid(u) \wedge aid(u') = aid(u) \ \wedge \ oid(u') = oid(u)\} \mid \geq 1$$

5 Conclusion

In this paper, we highlighted through representative usage scenarios the additional requirements that are posed when attempting to utilize the UCON family of models in modern computing environments. A classification of usage decision criteria, originating from either *direct* or *indirect* entities, highlighted the limitations of UCON model and spotted the necessity for a new use-based usage control model. UseCON model presented in this paper, supports complicated operations and eliminates the restrictions imposed by the attribute mutability mechanism regarding the utilization of historical information of usages. A number of examples were presented in order to demonstrate the enhanced capabilities of the UseCON model. The simplification of the policy administration process and the support of enhanced policy rules regarding their expressiveness, are included in the advantages of the proposed model. Moreover, the new characteristics of UseCON can be utilized by properly designed usage decision mechanisms in order to provide more sophisticated capabilities, as these are presented in the usage negotiation and actions hierarchy examples. The detailed investigation and analysis of the performance implications, if any, of the UseCONs modeling decisions is considered as future work.

References

1. Katt, B., Zhang, X., Breu, R., Hafner, M., Seifert, J.P.: A general obligation model and continuity: enhanced policy enforcement engine for usage control. In: Proceedings of the 13th ACM symposium on Access control models and technologies. pp. 123–132. SACMAT '08, ACM, New York, NY, USA (2008)
2. Lampson, B.W.: Protection. SIGOPS Oper. Syst. Rev. 8, 18–24 (January 1974)
3. Lazouski, A., Martinelli, F., Mori, P.: Usage control in computer security: A survey. Computer Science Review 4(2), 81 – 99 (2010)
4. OASIS: Oasis extensible access control markup language (xacml) tc (2011), http://www.oasis-open.org/
5. Park, J., Sandhu, R.: The ucon abc usage control model. ACM Trans. Inf. Syst. Secur. 7, 128–174 (February 2004)
6. Qiu, L., Zhang, Y., Wang, F., Kyung, M., Mahajan, H.R.: Trusted computer system evaluation criteria. In: National Computer Security Center (1985)
7. Samarati, P., Vimercati, S.D.C.d.: Access control: Policies, models, and mechanisms. In: Revised versions of lectures given during the IFIP WG 1.7 International School on Foundations of Security Analysis and Design on Foundations of Security Analysis and Design: Tutorial Lectures. pp. 137–196. FOSAD '00, Springer-Verlag, London, UK, UK (2001)
8. Sandhu, R.S.: Lattice-based access control models (1993)
9. Sandhu, R., Coyne, E., Feinstein, H., Youman, C.: Role-based access control models. Computer 29(2), 38 –47 (feb 1996)
10. Zhang, X., Parisi-Presicce, F., Sandhu, R., Park, J.: Formal model and policy specification of usage control. ACM Trans. Inf. Syst. Secur. 8, 351–387 (November 2005)
11. Zhang, X., Sandhu, R., Parisi-Presicce, F.: Safety analysis of usage control authorization models. In: Proceedings of the 2006 ACM Symposium on Information, computer and communications security. pp. 243–254. ASIACCS '06, ACM, New York, NY, USA (2006)