CA-UCON: a Context-Aware Usage Control Model

Abulgader Almutairi
Software Technology Research Laboratory
De Montfort University
The Gateway, Leicester, LE1 9BH, UK
Abdulgader@dmu.ac.uk

François Siewe
Software Technology Research Laboratory
De Montfort University
The Gateway, Leicester, LE1 9BH, UK
fsiewe@dmu.ac.uk

ABSTRACT

Usage CONtrol (UCON) model is the latest major enhancement of the traditional access control models which enables mutability of subject and object attributes, and continuity of control on usage of resources. In UCON, access permission decision is based on three factors: authorisations, obligations and conditions. While authorisations and obligations are requirements that must be fulfilled by the subject and the object, conditions are subject and object independent requirements that must be satisfied by the environment. As a consequence, access permission may be revoked (and the access stopped) as a result of changes in the environment regardless of whether the authorisations and obligations requirements are met. This constitutes a major shortcoming of the UCON model in pervasive computing systems which constantly strive to adapt to environmental changes so as to minimise disruptions to the user. To overcome this limitation, this paper proposes a Context-Aware Usage CONtrol (CA-UCON) model which extends the traditional UCON model to enable adaptation to environmental changes in the aim of preserving continuity of access. When the authorisations and obligations requirements are met by the subject and the object, and the conditions requirements fail due to changes in the environment or the system context, CA-UCON model triggers specific actions to adapt to the new situation. Besides the data protection, CA-UCON model so enhances the quality of services, striving to keep explicit interactions with the user at a minimum.

Categories and Subject Descriptors

D.1.6 [**Operating Systems**]: Security and Protection—*Access Controls*; F.1.2 [**Computation by Abstract Devices**]: Modes of Computation—*Interactive and reactive computation*; K.6.5 [**Management of Computing and Information Systems**]: Security and Protection—*unauthorized access*

Casemans '11, Sep 18, 2011, Beijing, China.

General Terms

Security, Human Factors

Keywords

Pervasive computing, context-aware, usage control, adaptive systems

1. INTRODUCTION

Information security is a key factor for the acceptance and adoption of pervasive computing systems where information can be accessed and shared by a multitude of small devices through wireless networks. Unless users are confident enough that information are exchanged securely, many would be deterred from using such systems. Nowadays, various portable devices such as smart phones, Personal Digital Assistants (PDAs) and laptop computers are being used to share information and to access digital resources via wireless connection to the Internet. Because these are resources constrained devices and highly mobile, changes in the environmental context (e.g. location and network availability) or device context (e.g. memory and battery) can affect the security of the system a great deal. A proper security mechanism must be put in place which is able to cope with changing environmental and system context.

Usage CONtrol (UCON) model [4] is the latest major enhancement of the traditional access control models which enables mutability of subject and object attributes, and continuity of control on usage of resources. The concept of mutability refers to the fact that attributes are not static but does change intermittently and hence the access permission decision should be dynamic and kept being reevaluated constantly when a new update occurs. Continuity of access decision ensures that decision to permit and allow access to an object is made constantly before and during the access to an object. This access decision is based on three key factors: authorisations, obligations and conditions. Authorisations are requirements on subject and object attributes that must hold for permission to be granted; obligations are mandatory requirements a subject has to perform before permission is granted; and conditions are requirements the environmental or system context must fulfil before access is permitted.

Because of the continuity of access decision, access permission may be revoked (and hence the access stopped) as a result of changes in the environmental or system context, regardless of whether the authorisations and obligations requirements are met. This constitutes a major shortcoming of the UCON model in pervasive computing systems which

constantly strive to adapt to environmental changes so as to minimise disruptions to the user. This paper proposes a Context-Aware Usage CONtrol (CA-UCON) model which extends the traditional UCON model to enable adaptation to environmental changes in the aim of preserving continuity of access. Indeed, when the authorisations and obligations requirements are met by the subject and the object, and the conditions requirements fail due to changes in the environmental or the system context, CA-UCON model triggers specific actions to adapt to the new situation. Besides the data protection, CA-UCON model so enhances the quality of services, keeping explicit interactions with the user at a minimum. Our contributions are summarised as follows:

- The architecture of a novel usage control model, CA-UCON, is proposed (Sect. 2); its main innovative feature is the integration of continuity of usage decision and dynamic adaptation to changes in the environmental or system context, so as to ensure continuity of usage.
- The computational model of the CA-UCON model is formally specified as a Finite State Machine (FSM) which describes how an access request is handled in the CA-UCON model (Sect. 3).
- The formal definitions of the CA-UCON$_{\text{ABD}}$ family core models are given (Sect. 4), where A stands for *Authorisations*, B for *oBligations* and D for *aDapatations*.
- Finally, we show that the UCON model can be specified in CA-UCON (Sect. 5) and so all the security models that can be specified in UCON, such as Role-Based Access Control (RBAC) and Digital Rights Management (DRM).

2. ARCHITECTURE OF CA-UCON MODEL

The architecture of the CA-UCON model is depicted in Fig. 1. It highlights the usage decision (UD) component and the adaptation decision (AD) component. The former is described as in the UCON model, making decision of granting or denying rights based on the authorisation, obligation and condition components. The latter decides what adaptation action to perform depending on the environmental context which is refined into subject context, object context, information and communication technology (ICT) context, and the physical environment context. The two dashed ovals materialise the fact that usage decision and adaptation decision happen continuously before and during usage. Following are the intuitive meanings of the key components of the CA-UCON model.

Subjects (S) and subject Attributes (ATT(S)).

A subject is an entity who requests access to a resource and must hold certain rights of access to the target object or resource. Subject has attributes which are used in the usage decision making. We let S denote the set of subjects and $ATT(S)$ denote the set of subject attributes.

Object (O) and Object Attributes (ATT(O)).

Object is the resource or entity which the subject has to hold a certain right to access or use. Object attributes are the descriptions and properties of a given object which

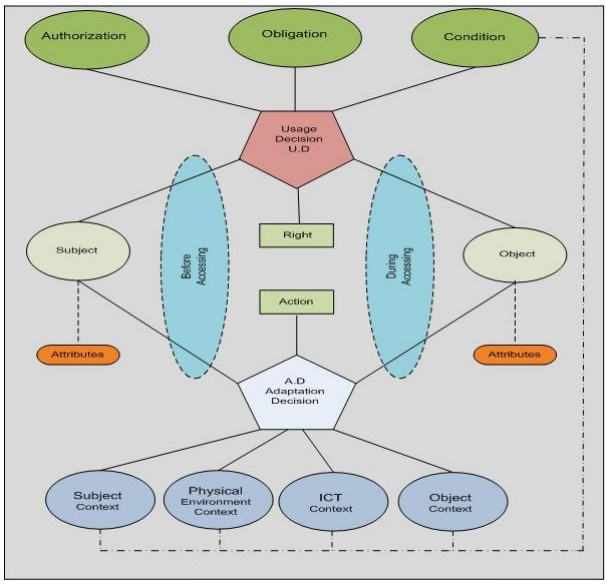

Figure 1: Architecture of The CA-UCON model

could be used as the basis for the provision and making of the usage decision process. Let O denote the set of objects and $ATT(O)$, the set of object attributes.

Rights (R).

Rights are privileges that subject can hold and use on an object. The subject must fulfil the authorizations, obligations and conditions requirements in order to be granted the right to access the object. The subject loses this right anytime one of these requirements does not hold. If this happen during access, there are two possibilities: (i) if either authorisations requirement or obligations requirement is not fulfilled, then the access right is revoked and the access stopped at once; (ii) if both authorisations requirement and obligations requirement are fulfilled, but the conditions requirement is not met due to changes in the environmental context, the system will attempt to adapt to the new situation by performing specific adaptation actions (including request to alternative object); if the adaptation is successful then the access continues, otherwise the access right is revoked and the access terminated.

Authorisations (A).

Authorisations are a key functional requirement that must be fulfilled before granting a particular right of access to a digital object. Authorization predicates place conditions and constraints in the form of logical predicates on both the subject and object attributes. The authorization predicates are activated and evaluated both before (pre-authorization) and during (ongoing-authorization) access.

oBligations (B).

Obligations are also functional predicates which are used to confirm mandatory requirements that a subject must undertake both before and during a particular usage process. The mandatory requirements here may be either pre-obligations ($preB$) to be fulfilled before access permission is granted or ongoing-obligations (onB) to be fulfilled during access.

Conditions (C).
Conditions are environmental constraints that must be considered in the process of usage decisions. Conditions are not related directly to objects or subjects, but they are based on environmental attributes. The evaluation of condition predicates may take place before granting permission to access a digital object (pre-conditions) or while the subject is using the object (on-conditions). When conditions fail due to changes in the environmental context, adaptation actions are triggered in an attempt to change the environmental context such that these conditions hold.

Subject Context.
Subject context is any type of context information linked to the subject such as his location, activity, preferences, and people nearby.

Object Context.
Object context refers to any kinds of context information related to the object. These can be the location of the object, execution state, nearby resources and availability.

Physical environments context.
This characterises relevant physical phenomena taking place such as the time, light, noise level, temperature, weather and so on.

ICT context .
ICT context is general term that deals with any kind of context information related to ICT and computing system included any communication devices or applications nearby. Examples of these contexts include: laptops battery rate, network reliability, smart phones memory size, PDAs and hardware capability and communication bandwidth. In addition, the diverse services and applications related to them, such as video-conferencing and distance learning.

Adaptation Actions .
Adaptation action is an operation that should be performed over condition predicate with the purpose of overcoming the environments changes. These actions may be classified according to the subject of the adaptation and the scope. For example, service instance adaptation actions (retry, duplicate service, and substitute service) and flow instance adaptation actions (redo, choose alternative service, and undo).

3. COMPUTATIONAL MODEL OF THE CA-UCON MODEL

The computational model of CA-UCON model can be described as a Finite State Machine (FSM) depicting how an subject's request to access an object is handled in the CA-UCON model. The FSM is depicted by the graph in Fig. 2, where nodes are called *states* and edges are called *transitions*. The initial state, labelled *initial*, corresponds to the state when the system is waiting for a suject to submit a request. There are three final states: *end*, when the access has successfully terminated; *denied*, when the access request has been denied; and *revoked*, when access permission has been revoked during access and hence the access stopped. The intuitive meaning of the remaining states of the FSM can be summarised as follows: *requesting*, denotes when the access request is being processed; *accessing*, represents the state when the actual access is taking place; *preadapting*, is the state when the system is trying to adapt to the environmental context prior to access; and finally *onadapting*, is when the system is trying to adapt to the environmental context during access.

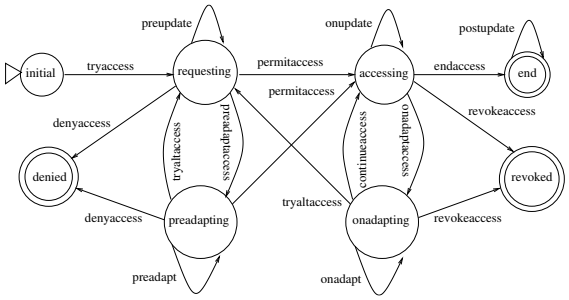

Figure 2: Execution of an access request in the CA-UCON model

The transitions of the FSM are labelled with the events (or actions) that fire them. The event *tryaccess* occurs when a suject sends an access request (e.g. by clicking a menu button). This event forces the FSM to enter the *requesting* state to process that access request. While in this state, the system can perform updates on subject's and object's attributes through *preupdate* events. If the authorisations, obligations and conditions requirements are all met, the system emits the *permitaccess* event and moves into the *accessing* state. If for some reasons either the authorisations requirement or the obligations requirement is not met, the sytem emits the event *denyaccess* and terminates in the *denied* state. However, if both the authorisations requirement and obligations requirement are met, but the conditions requirement is not satisfied, the system emits the *preadaptaccess* event and moves into the *preadapting* state. In this state, specific adaptation actions, denoted by the *preadapt* events, are performed in an attempt to meet the conditions requirement. If the adaptation is successful, the *permitaccess* event is raised and the system transitions into the *accessing* state. In addition, a new request to access a specified alternative object, denoted by the *tryaltaccess* event, may be issued automatically by the system if the adaptation actions fail. Otherwise the access request is simply denied when no adaptation is possible.

When access permission is granted (see *permitaccess* event), the system transitions into the *accessing* state in which the actual access takes place. During access the system can perform updates on subject's and object's attributes via *onupdate* events. If during access either the authorisations requirement or the obligations requirement is not met, the sytem emits the event *revokeaccess* and terminates in the *revoked* state. However, if both the authorisations requirement and obligations requirement are continuously met, but the conditions requirement fails, the system raises the *onadaptaccess* event and moves into the *onadapting* state. In this state, specific adaptation actions, denoted by the *onadapt* events, are performed in an attempt to meet the conditions requirement. If the adaptation is successful, the *continueaccess* event is raised and the system moves back into the *accessing* state. In the effort to enhance the quality of service even further, the system might issue an implicit re-

quest to access a specified alternative object through the *tryaltaccess* event, when the adaptation actions fail. In the worst case when no adaptation is possible, the access permission is simply revoked and the access stopped at once.

When an access terminates successully via the *endaccess* event, the system moves into the *end* state and eventually performs updates on subject's and object's attributes through *postupdate* events.

4. THE CA-UCON$_{\text{ABD}}$ FAMILY CORE MODELS

Park et al. [4] defined the UCON$_{\text{ABC}}$ family core models where A stands for Authorisations, B for oBligations and C for Conditions. Here we define the CA-UCON$_{\text{ABD}}$ family core models where C is replaced by D for aDaptation. So the CA-UCON$_{\text{A}}$ and CA-UCON$_{\text{B}}$ family core models are identical to UCON$_{\text{A}}$ and UCON$_{\text{B}}$, respectively. The CA-UCON$_{\text{D}}$ family core model comprises two models: the pre-adaptation model CA-UCON$_{\text{preD}}$ and the ongoing adaptation model CA-UCON$_{\text{onD}}$, which are detailed below.

4.1 The CA-UCON$_{preD}$ Model

In the CA-UCON$_{preD}$ model, adaptation can be activated only before the access permission is granted. That is adaptation cannot take place during access. If s is subject, o and object and r an access right, we let $preD(s, o, r)$ denote a predicate which is true if the pre-adaptation is successful and false otherwise. We also denote the access permission decision by the predicate $allowed(s, o, r)$. The CA-UCON$_{preD}$ core model is composed of the following elements:

- S: set of subjects, *ATT(S)*: set of subject attributes, O: set of objects, *ATT (O)*: set of object attributes
- AD: set of adaptation actions.
- $PreCON$: set of pre-conditions elements.
- T: time domain.
- $PreAdapted : 2^{preCON} \times AD \times T \longrightarrow \{true, false\}$

  preAdapted(c, a, t) is a boolean function that performs the adaptation action a until all the conditions in c evaluate to true, in which case the function returns *true*; otherwise the function returns *false* after t time-units have elapsed since the execution of the action a started.
- $getPreADAPT : S \times O \times R \longrightarrow 2^{preCON} \times AD \times T$

  getPreADAPT(s, o, r) returns a tuple *(c, a, t)* where c is the set of all pre-conditions required to grant the subject s the access right r upon the object o, a is the adaptation action to be performed if any of the pre-conditions does not hold, and t is the time-out for this adaptation process.
- $getPreAltReq : S \times O \times R \longrightarrow 2^{O \times R}$

  getPreAltReq(s, o, r) denotes the set of alternative requests that can be made on behalf of the subject s when the initial request of the access right r upon the object o could not be granted due to environmental conditions.
- *preD(s, o, r) = preAdapted(getPreADAPT(s, o, r))*
- The access permission decision is defined as:

$$allowed(s, o, r) \Rightarrow \left(\begin{array}{c} preD(s, o, r) \\ \vee \\ \bigvee\limits_{(o', r') \in E} allowed(s, o', r') \end{array} \right)$$

  where $E = getPreAltReq(s, o, r)$ and the symbol '$\Rightarrow$' denotes the logical implication.

4.2 The CA-UCON$_{onD}$ Model

In the CA-UCON$_{onD}$ model, there is no pre-adaptation; adaptation can only take place during access. If s is subject, o and object and r an access right, we let $onD(s, o, r)$ denote a predicate which is true if the ongoing adaptation is successful and false otherwise. We also denote by $stopped(s, o, r)$ a predicate which is true if the access has been stopped. The CA-UCON$_{onD}$ core model is composed of the following elements:

- S: set of subjects, *ATT(S)*: set of subject attributes, O: set of objects, *ATT (O)*: set of object attributes
- AD: set of adaption strategies (or actions)
- $onCON$: set of ongoing-conditions elements
- T: time domain
- $onAdapted : 2^{onCON} \times AD \times T \rightarrow \{true, false\}$

  onAdapted(c, a, t) is a boolean function that performs action a until all the conditions in c evaluate to *true*, in which case the function returns *true*; otherwise the function returns *false* after t time-units have elapsed since the execution of the action a started.
- $getOnADAPT : S \times O \times R \rightarrow 2^{onCON} \times AD \times T$

  getOnADAPT(s,o, r) returns a tuple *(c, a, t)* where c is the set of all ongoing-conditions required for the subject s to keep the right r upon the object o during access, a is the adaptation action to be performed if any of the ongoing-conditions does not hold, and t is the time-out for this adaptation process.
- $getOnAltReq : S \times O \times R \rightarrow 2^{O \times R}$

  getOnAltReq(s, o, r) denotes the set of alternative requests that can be made on behalf of the subject s when the initial request of the access right r upon the object o fails during access due to environmental conditions.
- *onD(s, o, r) = onAdapted(getOnADAPT(s, o, r))*
- *allowed(s, o, r) $\Rightarrow$ true*
- The predicate *stopped* is defined as follows:

$$stopped(s, o, r) \Leftarrow \left(\begin{array}{c} \neg onD(s, o, r) \\ \wedge \\ \bigwedge\limits_{(o', r') \in F} stopped(s, o', r') \end{array} \right)$$

  where $F = getOnAltReq(s, o, r)$ and the formula $V \Leftarrow W$ means that W implies V.

5. EXPRESSIVE POWER OF THE CA-UCON MODEL

In this section we show that the UCON model can be specified in CA-UCON and so all the security models that can be specified in UCON, such as Role-Based Access Control (RBAC) and Digital Rights Management (DRM). As mentioned in the previous section, the authorisation and obligation family core models of CA-UCON are identical to those of UCON. Rest to prove that the condition family core models of UCON can be modelled by the adaptation family core models of CA-UCON.

Indeed, the UCON$_{preC}$ model is a special case of CA-UCON$_{preD}$ model where:

- $AD = skip$, where {skip} is a special action that does nothing and lasts one time-unit.
- $T = \{1\}$, the unique time-out is one time-unit.
- *getPreAltReq(s, o, r)* $= \phi$, for all $(s, o, r) \in S \times O \times R$.

Similarly, the UCON$_{onC}$ model is a special case of CA-UCON$_{onD}$ model where:

- $AD = \{skip\}$, where skip is a special action that does nothing and lasts one time-unit.
- $T = \{1\}$, the unique time-out is one time-unit.
- *getOnAltReq(s, o, r)* $= \phi$, for all $(s, o, r) \in S \times O \times R$.

6. RELATED WORK

Many works have been carried out in the area of context-aware access control model which combine the context information with credentials while making access control decisions. A context-aware role-based access control (CGRBAC) model was proposed by [6], to address a new set of challenges which was not addressed by the traditional security models, hence introducing "global rol" and "contex" to the basic RBAC model. The model can be expanded to address global services and as well as environment-relevant issues. Moreover, another model was proposed by [2] in which they generalized the context-based access control model that offers the resource to owners and access control administrators the ability of defining the context-based access policies considering the context of the information describing the owner's, requester's and resource's situations. The model also considers both the owner's, requestor's and resource's context when making context-based access control decisions. However, the proposed model extends its support for defining access policies which are completely based on the context information, outlining seven types of context-based access context control policies.

[7] came out with the idea of control architecture, the context-aware access control enable the provision of e-services based on an end-to-end web services infrastructure. Furthermore, the proposed control architecture can allow control access to distribute web services through an intermediary server, transparently to both clients and protected resources. This access control mechanism is based on an RBAC model which incorporates dynamic context information. Although, [1] has extended the traditional role-based access control that imbeds the notion of an environment role, their approach focuses on solving the problem of accessing dynamic context services and utilizing environment roles for the user-aware web in a context aware computing environment. It can be noticed that the research has shown how the developed concept of a role can be implemented to capture relevant security in the context of the environment in which access requests are made.

[3] suggested that an adaptive access control scheme should be utilizing a context awareness in pervasive computing environments in which he designed an adaptive access control model based on the traditional RBAC model, and also presented an adaptive access control scheme to guarantee flexibility to the user and according to the changes of context. According to [8] a Context-aware Task-role based Access Control (CTRBAC) model should provide a detail context-aware access control for pervasive computing in enterprise environments. The ideal behind it, is that it extends the T-RBAC model by dynamically adjusting the role assignments based on the current context information and ultimately deciding on whether the user is authorized to execute the task or not.

[5] also mentioned that a context aware access control model based on the RBAC model . As such the model can assign roles dynamically to the users and restrict their access within the context of information. The model was presented in a formal simple case study to demonstrate the application of the model.

However, the above-mentioned research works do apply context-awareness on traditional access control such as Role-based Access Control (RBAC). In our work, we extend the usage control model (UCON), which is the latest major enhancement of the traditional access control models, to make it adaptive and context-aware.

7. CONCLUSION

In this paper we proposed a context-aware usage control model (CA-UCON) that extends the traditional UCON model to enable adaptation to environmental condition (or context). In addition to the authorisation and obligation family core model of UCON, CA-UCON includes two new core models: the pre-adaptation model (CA-UCON$_{preD}$) and the ongoing adaptation model (CA-UCON$_{onD}$). We show that these two models can be used to represent the pre-condition model and ongoing-condition model of UCON, respectively.

In future works, we will investigate a formal specification of the CA-UCON model and investigate possible enforcement mechanisms of the model in a pervasive environment.

8. REFERENCES

[1] L.-C. F. Chun-Dong Wang, Ting Li. Context-aware environment-role-based access control model for web services. In *2008 International Conference on Multimedia and Ubiquitous Engineering*, 2008.

[2] H. M. José Bringel Filho. A generalized context-based access control model for pervasive environments. In *Proceedings of the 2nd SIGSPATIAL ACM GIS 2009 International Workshop on Security and Privacy in GIS and LBS*, 2009.

[3] H. J. Jung Hwan Choi, Dong Hyun Kang and Y. I. Eom. Adaptive access control scheme utilizing context awareness in pervasive computing environments. In *Performance, Computing and Communications*

Conference, 2008. IPCCC 2008. IEEE International, 2008.

[4] J. Park and R. Sandhu. The UCON$_{ABC}$ usage control model. *ACM Transactions on Information and System Security*, 7(1):128–174, February 2004.

[5] S. Z. Sareh Sadat Emami, Morteza Amini. A context-aware access control model for pervasive computing environments. In *2007 International Conference on Intelligent Pervasive Computing*, 2007.

[6] H. F. SHEN Haibo. A context-aware role-based access control model for web services. In *Proceedings of the 2005 IEEE International Conference on e-Business Engineering (ICEBEÂŠ05)*, 2005.

[7] D. K. S. K. Vassilis Kapsalisa, Loukas Hadellisb. A dynamic context-aware access control architecture for e-services. *Computers & Security*, 25:507–521, 2006.

[8] R. X. Zhou Zhu. A context-aware access control model for pervasive computing in enterprise environments. In *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, 2008.