XCON
Internet-Draft
Expires: October 18, 2004

H. Khartabil
P. Koskelainen
Nokia

April 19, 2004

The Conference Policy Control Protocol (CPCP)
draft-ietf-xcon-cpcp-xcap-00

Status of this Memo

Copyright Notice

Abstract

This document describes the Conference Policy Control Protocol (CPCP). It specifies an Extensible Markup Language (XML) Schema that enumerates the conference policy data elements that enable a user to define a conference policy. It also defines an XML Configuration

Access Protocol (XCAP) application usage that is needed to store and manipulate a conference policy.

Table of Contents

1.   Introduction   3
2.   Conventions Used in This Document   3
3.   Terminology   3
4.   Structure of a Conference Policy document   4
4.1   MIME Type for CPCP XML Document   6
4.2   XML Document Description   6
4.2.1   <Conference-settings> element   6
4.2.2   <Conference-info> element   6
4.2.3   <Conference-time> element   7
4.2.4   <ACL> (Access Control List) element   8
4.2.5   <PCL> (Privilege Control List) element   9
4.2.6   <DL> (Dial-Out List) element 10
4.2.7   <SC> (Security Control) element 10
4.2.8   <Conference-floor-policy> element 11
4.2.9   <Conference-media-Policy> element 12
4.3   XML Schema 12
5.   Floor Control Policy vs. Floor Control Protocol 19
6.   An XCAP Usage for Conference Policy Manipulation 19
6.1   Application Unique ID 19
6.2   Resource Interdependencies 19
6.3   Additional Constraints 20
6.4   Naming Conventions 20
6.5   Authorization Policies 20
6.6   MIME Type for CPCP XML Document 20
6.7   Overview of Operation 20
6.8   Communication Between Conference Entities 21
6.9   Conference Creation and Termination 21
6.10   Manipulating the Participant Lists 21
6.10.1 Expelling a Participant 22
6.11   Privileges: Who can modify the conference policy 22
6.12   Conference URI(s) 23
7.   Examples 23
8.   Security Considerations 26
9.   IANA Considerations 26
9.1   XCAP Application Usage ID 26
9.2   application/conference-policy+xml mime TYPE 26
9.3   URN Sub-Namespace Registration for
        urn:ietf:params:xml:ns:conference-policy 27
10.   Contributors 28
11.   Acknowledgements 28
        Normative References 28
        Authors' Addresses 29
        Intellectual Property and Copyright Statements 31

1. Introduction

SIP conferencing framework [11] defines the mechanisms for multi-party centralized conferencing in a SIP environment. Existing SIP mechanisms allow users, for example, to join and leave a conference. A centralized serve, called focus, can expel and invite users, and may have proprietary access control lists and user privilege definitions. However, in many cases it is useful to have a standardised conference policy elements such as access control lists and a standardised protocol means to manipulate them. The requirements for such protocol are defined in [7]. This document provides an XML Schema Section 4.3 that enumerates the conference policy data elements that enable a user to define a conference policy. It also defines an XML Configuration Access Protocol (XCAP) [8] application usage that is needed to store and manipulate a conference policy.

Other mechanisms, such as web page or voice response system can also be used to manipulate conference policy data.

XCAP has many advantages in its use for conference policy control protocol. It is a HTTP 1.1 based protocol that allows clients to read, write, modify and delete application data stored in XML format at a server. XCAP maps XML document elements and attributes to HTTP URIs that can be directly accessed by HTTP. One application area which has already adopted XCAP is the manipulation of event lists [9].

2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

3. Terminology

This document uses terminology from [11]. Some additional definitions are introduced here.

ACL

Access control list (ACL) defines users who can join to a conference. Users may have allowed, blocked, pending or expelled status in the list. Each conference has its own ACL.

CPS

Conference Policy Server. See [11]

Conference participant

Conference participant is a user who has on-going session (e.g. SIP dialog) with the conference focus.

Floor control

Floor control is a mechanism that enables applications or users to gain safe and mutually exclusive or non-exclusive access to the shared object or resource in a conference.

Dial-Out List (DL)

Dial-out list (DL) is a list of users who the focus needs to invite to the conference.

PCL

Privilege control control (PCL) defines privileges for a user. Each user in a conference may have different list of privileges and each conference has its own PCL.

Privileged user

In this document, a privileged user is the creator. Defining privileges to modify certain parts of a conference policy is outside the scope of this document.

CPS XCAP URI

The URI of the XCAP server that is used to create the conference. The URI contsruction is specified in [8]. It is refered to in XCAP as the host part.

Conference Policy URI

The URI of conference policy. In XCAP, it is the CPS XCAP URI along with the abs_path. It identifies the XML document. The URI contsruction is specified in [8].

4. Structure of a Conference Policy document

The conference policy document is an XML [5] document that MUST be well-formed and MUST be valid. Conference policy documents MUST be based on XML 1.0 and MUST be encoded using UTF-8. This specification makes use of XML namespaces for identifying conference policy documents and document fragments. The namespace URI for elements defined by this specification is a URN [2], using the namespace identifier 'ietf' defined by [3] and extended by [13]. This URN is:

urn:ietf:params:xml:ns:conference-policy

A conference policy document begins with the root element tag "conference-policy". Other elements from different namespaces MAY be present for the purposes of extensibility. Elements or attributes from unknown namespaces MUST be ignored. The conference policy is build up using multiple namespaces:

- "urn:ietf:params:xml:ns:conference-settings": This namespace defines elements for conference setting. The inclusion of this namespace is optional. It contains the mandatory element <Conference-settings>. This element contains the conference URI(s) and maximum number of participants. It can occur only once in the document.

- "urn:ietf:params:xml:ns:conference-info": This namespace defines elements to carry conference information. The inclusion of this namespace is optional. It contains the mandatory element <Conference-info>. This element includes informational describing the conference, e.g. for search purposes. This information can also be used in the session description when the focus is sending invitations. It can occur only once in the document.

- "urn:ietf:params:xml:ns:conference-time": This optional namespace defines conference time information. It defines the mandatory <Conference-time> element that includes elements defining start and stop times for a conference.

- "urn:ietf:params:xml:ns:conference-acl": This optional namespace is for the access control list. It defines the mandatory <ACL> element that contains URIs for users who can dial into the conference, users who are blocked from dialling in, and expelled users.

- "urn:ietf:params:xml:ns:conference-pcl": This optional namespace is for the privilege control list. It defines the mandatory <PCL> element that contains privileges and URIs for users who have those privileges.

- "urn:ietf:params:xml:ns:conference-dl": This optional namespace is for the dial-out list. It defines the mandatory <DL> element that contains URIs for users that the focus will invite to the conference.

- "urn:ietf:params:xml:ns:conference-sc": This optional namespace is for security control. It defines the <SC> element that contains conference security level and passwords.

- "urn:ietf:params:xml:ns:conference-mp": This optional namespace is for the media policy for a conference. It defines the <Conference-media-policy> element that contains the media types to be used in the conference.

- "urn:ietf:params:xml:ns:conference-fp": This optional namespace is for the floor control policy. It defines the <Conference-floor-policy> element.

The elements are described in more detail in the forthcoming sections.

4.1 MIME Type for CPCP XML Document

The MIME type for the CPCP XML document is "application/conference-policy+xml".

4.2 XML Document Description

4.2.1 <Conference-settings> element

The mandatory <Conference-settings> element contains 2 sub-elements; the <Conference-URI> element and the <Max-participant-count> element.

<Conference-URI> is an optional element. It can occur more than once to accommodate multiple signaling protocols. Once a conference URI is set, it MUST NOT be changed or removed for the duration of the conference. Only one URI per protocol MUST be set. URIs can be added at any time.

This is in its own XML namespace, so it is separated from other elements and hence relevant modification rights (privileges) can be given more easily to other namespaces.

<Max-participant-count> is an optional. It carries the maximum number

of participants allowed in the conference. When the maximum number of participants threshold is reached, no new users are not allowed to join until the number of participants decreases again. If using SIP, the server can reject a request to join (INVITE) with a "480 Temporarily Unavailable" response. Alternatively, the sever may implement a waiting queue.

4.2.2 <Conference-info> element

Mandatory <Conference-info> element has its own namespace and it can occur only once in a document. It includes informative conference parameters which may be helpful describing the purpose of a conference, e.g. for search purposes or for providing host contact information. The <Conference-info element MUST have a special attribute 'xml:lang' to specify the language used in the contents of this element as defined Section 2.12 of [5].

Each conference has an optional <Subject> element, which describes the current topic in a conference. The optional <Display-name> element is the display name of the conference, which usually does not change over time.

<Free-text> and <Keywords> are optional elements. They provide additional textual information about the conference. This information can be made available to potential conference participants by means outside the scope of this document. Examples of usage could be searching for a conference based on some keywords. The optional <Web-page> element points to a URI where additional information about the conference can be found.

The optional <Host-info> element contains several elements. It gives additional information about the user hosting the conference. This information can, for example, be included into the SDP fields of the SIP INVITEs sent by the focus. The <URI> element is optional and can occur more than once.

4.2.3 <Conference-time> element

The information related to conference time and lifetime is contained in the <Conference-time> element. The conference may occur for a limited period of time (i.e. bounded), or the conference may be unbounded (i.e. it does not have a specified end time). Bounded conferences may occur multiple times(e.g. on weekly basis).

<Conference-Time> has its own XML namespace. It contains one or more <Conference-occurrence> elements each defining the time information of a single conference occurrence. Multiple <Conference-occurrence> elements MAY be used if a conference is active at multiple irregularly spaced times; each additional <Conference-occurrence> element contains time information for a specific occurrence.

For each occurrence, the <Start-time> element specifies when a conference starts. the <Stop-time> element specifies the time a conference stops. If the <Start-time> element is not present, it indicates that the conference starts immediately. If the <Stop-time> is set to zero, then conference occurrence is not bounded, i.e. permanent, though it will not become active until the <Start-time>. If the <Stop-time> element is not present, it indicates that the conference terminates as soon as the last participant leaves the conference. The focus might wait a small period of time before terminating the conference, in case a participant joins straight after the last participant leaves.

When saying that a conference starts, or becomes active (start-time), it means that the mixing starts. A focus will most likely allow participants to connect shortly before start time, but may put them on hold until the start time. Participants on the Dial out list may also be dialled to shortly before start time.

A conference terminates with stop-time. The creator is free to set the stop-time to be the time s/he leaves (and therefore the conference terminates when s/he leaves), terminate the conference as s/he leaves (modifying stop-time), or leave before the stop-time and therefore the conference continues. The stop-time can be changed by the conference creator, during the conference, to allow the extension of the conference based on best effort. A conference always terminates when the conference policy is removed, regardless of the stop time.

The absence of this conference time information indicates that a conference starts immediately and terminates when the conference policy is removed. See Section 6.9 for more details

4.2.4 <ACL> (Access Control List) element

ACL has its own XML namespace.

The purpose of Access Control List (ACL) is to control who can join the conference.A conference has one <ACL> consisting of one or more <ACL-target-URI> elements and the <Access-type> parameter for those URIs. Access-Types are one of Allowed/Blocked/Pending/Expelled. Allowed means that the   target is allowed to join the conference. Blocked means that the target is not allowed to join the conference. This can be used in the where the allowed URIs are wild-carded and the user wants to explicitly block one potential participant, whose URI falls within the wildcarded URIs, from joining. The other way around is also possible where the blocked URIs are wildcarded and the user wants to explicitly allow one potential participant, whose URI falls within the wildcarded URIs, to join. Pending means that authorisation for the target is not granted and while further processing is required - such as consulting the moderator. Expelled means   that user is expelled from current conference and is not allowed to join or be dialled-out (even if dial-out list includes user's URI).

Wildcards are allowed in ACL as follows. The domain part is allowed to be wildcard only if the username is a wildcard. Wildcard in the domain part MUST be immediately after the @-sign. A wildcard in the domain is interpreted as multiple zones. For example: sip:*@*.example.com includes sip:*@engineering.example.com as well as sip:*@tester.engineering.example.com. The use of wildcarding has been restricted to avoid ambiguous entries in the access control list.

Examples of allowed wildcards are - sip:*@example.com, *@*.com, *@*.

Examples are not allowed wildcards are - sip:bob@example.*, sip:bob@*.com, sip:*@example.*.com.

"Most-specific expression wins" policy is used if overlapping rules are found. Basically, this means that user specific rule is searched first and if it is not found, then most specific wildcard rule is utilized.

There is a need for the ACL to contain an entry that defines the default access types for users not explicitly allowed nor blocked from joining the conference, i.e. everybody else. For example: "Pending" action for *@*. If that entry is missing, the focus local policy dictates the behaviour.

Sip: and sips: schemes are treated as equivalent in the ACL since it defines users and not the security used by users.

It is also possible to ask the focus to refer users to the conference. An optional Boolean attribute "refer" exists in the <ACL-target-URI> that indicates to the server that the creator of the conference wishes for the focus to refer the identified potential participants to the conference when a conference occurrence has started. In SIP, this is achieved by the focus sending a REFER request to those potential participants. The default value for the "refer" attribute is "false".

4.2.5 <PCL> (Privilege Control List) element

Advanced privilege models can be applied in conferencing context (e.g. who is allowed to modify ACL, who is allowed to expel users etc). This document defines only one privilege and leaves the definition of additional privileges (e.g. who can modify ACL) as a separate standardisation effort.

The <PCL> element is mandatory and has its own XML namespace. It defines which users has what privileges. The <PCL> element may contain one or more <PCL-target> elements. The <PCL-target> element carries 2 pieces of information: the target URI, <PCL-target-URI> and the privileges for that URI, <Privileges>. All mandatory elements.

The target URI can be wildcarded as described for the ACL in Section 4.2.4.

Example URIs are:

sip:bob@company.com

sip:*@example.com

The only privilege defined in this document is RIGHT_TO_SUBSCRIBE_TO_CONF_EVENT_PACKAGE. It defines which users are allowed to   subscribe to the conference state event package [12]and be notified.

4.2.6 <DL> (Dial-Out List) element

The dial-out list (DL) is a list of user URIs that the focus uses to learn who to invite to join a conference. This list can be updated during the conference lifetime so it can be used for mid-conference invites (and mass-invites) as well.

DL has its own XML namespace.

The <DL> element includes a mandatory <DL-target> element. The <DL-target> element includes the mandatory <DL-target-URI> element. This elements carries the URI of the user to be invited.

4.2.7 <SC> (Security Control) element

The conference security encompasses three aspects: controlling the visibility of a conference, securing the SIP messages, and performing authentication for individual users.

This element has its own XML namespace.

The conference security settings start with the mandatory >SC> element. It contains the mandatory <Visibility> element. This element can hold one of two values: visible or invisible. The <Visibility> element controls whether information in the <Conference-URI>, <Conference-time> and <Conference-info> elements may be made available publicly. For example, an application at a conference server might list the ongoing conferences on web page, or it may allow searching for conferences based on the keywords listed in the <Conference-info> element. Setting <Visibility> to "invisible" instructs the application not to reveal any such information. However, information in other elements, such as <ACL>, should not be seen by anyone else other than a privileged user, even with the <Visibility> element set to "visible".

We define two mechanisms for securing the signaling between users and the focus: TLS and S/MIME. TLS is used to provide transport layer security on a hop-by-hop basis. According to SIP [6], using SIPS URI scheme in a request signifies that TLS must be used to secure each hop over which the request is forwarded until the request reaches the SIP entity responsible for the domain portion of the Request-URI.

The <Security-mechanism>element inside the <SC> element has 2 boolean parameter: TLS and S/MIME. When in TLS parameter is set to "true" (thus implying the use of SIPS URI scheme, if SIP is used as the signaling protocol), it is required that TLS is used end-to-end. In other words, TLS must be used also on the last hop between the entity responsible for the domain portion of the Request-URI and the conference policy server.

If end-to-end confidentiality of entire SIP messages is not required by the conference policy, but it is required that the message bodies within SIP are encrypted, the S/MIME attribute must have a value "true".

TLS and S/MIME may be required independent of each other. In other words, it may be required to use neither, one, or both depending on the settings of these parameters.

The conference creator can define an authentication policy for the participants. This is done with the optional <SC-target> element.

If the <SC-target> element is present, then at least one <SC-target-URI> inside the <SC-target> element must be present, each identifies a user or a set of users for which the authentication mechanism apply. The target URI can be wildcarded as described for the ACL in Section 4.2.4.

The authentication policy defined in the optional <Authorization-mechanism> element defines how the participants should be authenticated. Two authentication mechanisms are defined in this document: Digest and Digest-AKA. The authentication policy can also be set to none. The password associated with each user in the Digest authentication is included in the optional <Password> attribute. This attribute is ignored if authentication is set to "none".

4.2.8 <Conference-floor-policy> element

This element has its own XML namespace. The absence of this namespace and its elements from an XML document indicates that the conference does not have a floor.

The <Conference-floor-policy> is mandatory and contains the required boolean attribute that indicates if the floor is moderator controlled or not. One or more <Floor> elements can appear in the <Conference-floor-policy> element. The number of those elements indicates how many floors the conference can have. A floor can be used for one or more media types; the mandatory <Media-types> element can contain zero or more of the <Video>, <Audio>, <Application>, <Data> ,<Control>, <Message>, and <text> elements indicating the media of the floor. One type of media can only appear once. Other media types can be defined by extensions.

A floor can be controlled using many algorithms; the mandatory <Algorithm> element MUST contain one and only of the <Moderator-controlled>, <FCFS>, and <Random> elements indicating the algorithm.

The <Max-floor-users> element in the <Floor> element is optional and, if present, dictates the maximum number of users who can have the floor at one time. The optional <Moderator-URI> indicates the URI of the moderator. It MUST be set if the attribute moderator-controlled is set to "true".

4.2.9 <Conference-media-Policy> element

Media policy is an integral part of the conference policy. It defines e.g. what kind of media topologies exist in the conference. This document defines a very basic media policy that states the media types a conference has. This is used by the focus to know what media types to invite users with and what media types it should accept from dialling in users. The details of media manipulation are defined elsewhere. User with sufficient privileges is allowed to create, modify and delete the media policy (e.g. add new media types).

This element has its own XML namespace.

The definition starts with the mandatory <Conference-media-policy> element. This element contains a mandatory <Media-types> element that lists the media types allowed for this conference. The format of this mirrors that of the same element in floor policy.

4.3 XML Schema

```
   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
                        targetNamespace="urn:ietf:params:xml:ns:conference-
policy"
                        xmlns:conference-mp="urn:ietf:params:xml:ns:conference-
mp"
```

```
                        xmlns:conference-fp="urn:ietf:params:xml:ns:conference-
fp"
                        xmlns:conference-sc="urn:ietf:params:xml:ns:conference-
sc"
                        xmlns:conference-dl="urn:ietf:params:xml:ns:conference-
dl"
                        xmlns:conference-
pcl="urn:ietf:params:xml:ns:conference-pcl"
                        xmlns:conference-
acl="urn:ietf:params:xml:ns:conference-acl"
                        xmlns:conference-
time="urn:ietf:params:xml:ns:conference-time"
                         xmlns:conference-
info="urn:ietf:params:xml:ns:conference-info"
                         xmlns:conference-
settings="urn:ietf:params:xml:ns:conference-settings"
                         elementFormDefault="qualified">
        <xs:import namespace="urn:ietf:params:xml:ns:conference-settings"
schemaLocation="conference-settings.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-info"
schemaLocation="conference-info.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-time"
schemaLocation="conference-time.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-acl"
schemaLocation="conference-acl.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-pcl"
schemaLocation="conference-pcl.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-dl"
schemaLocation="conference-dl.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-sc"
schemaLocation="conference-sc.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-fp"
schemaLocation="conference-fp.xsd"/>
        <xs:import namespace="urn:ietf:params:xml:ns:conference-mp"
schemaLocation="conference-mp.xsd"/>
        <xs:element name="Conference">
                <xs:complexType>
                        <xs:sequence>
                                <xs:element name="Conference-Settings"
type="conference-settings:conference-settings"/>
                                <xs:element name="Conference-Info"
type="conference-info:Conference-Info"/>
                                <xs:element name="Conference-Time"
type="conference-time:Conference-Time"/>
                                <xs:element name="ACL" type="conference-
acl:Conference-ACL"/>
                                <xs:element name="PCL" type="conference-
```

```
pcl:Conference-PCL"/>
                                <xs:element name="DL" type="conference-
dl:Conference-DL"/>
                                <xs:element name="SC" type="conference-
sc:Conference-SC"/>
                                <xs:element name="Conference-floor-policy"
type="conference-fp:Conference-Floor-Policy"/>
                                <xs:element name="Conference-media-policy"
type="conference-mp:Conference-Media-Policy"/>
                        </xs:sequence>
                </xs:complexType>
        </xs:element>
   </xs:schema>



   <!-- Conference settings -->


   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-settings"
                         xmlns="urn:ietf:params:xml:ns:conference-settings"
                         xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
        <xs:complexType name="Conference-settings">
                <xs:sequence>
                        <xs:element name="Conference-uri" type="xs:anyURI"
minOccurs="0" maxOccurs="unbounded"/>
                        <xs:element name="Max-participant-count"
type="xs:nonNegativeInteger" minOccurs="0"/>
                </xs:sequence>
        </xs:complexType>
   </xs:schema>



   <!-- Conference Info -->


   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-info"
                        xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">


             <!-- This import brings in the XML language attribute xml:lang-->
        <xs:import namespace="http://www.w3.org/XML/1998/namespace"
          schemaLocation="http://www.w3.org/2001/xml.xsd"/>


        <xs:complexType name="Conference-info">
                <xs:sequence>
                        <xs:element name="Subject" type="xs:string"
minOccurs="0"/>
                        <xs:element name="Display-name" type="xs:string"
minOccurs="0"/>
                        <xs:element name="Free-text" type="xs:string"
minOccurs="0"/>
                        <xs:element name="Keywords" minOccurs="0">
                                <xs:simpleType>
                                        <xs:list itemType="xs:string"/>
                                </xs:simpleType>
                        </xs:element>
                        <xs:element name="Web-page" type="xs:anyURI"
minOccurs="0"/>
                        <xs:element name="Host-info" minOccurs="0">
                                <xs:complexType>
                                        <xs:sequence>
                                                <xs:element name="URI"
type="xs:anyURI" minOccurs="0"/>
                                                <xs:element name="E-mail"
type="xs:anyURI" minOccurs="0"/>
                                                <xs:element name="Web-page"
type="xs:anyURI" minOccurs="0"/>
                                        </xs:sequence>
                                </xs:complexType>
                        </xs:element>
                </xs:sequence>
                <xs:attribute ref="xml:lang"/>
        </xs:complexType>
   </xs:schema>



   <!-- Conference time -->


   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-time"
                        xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
        <xs:complexType name="Conference-Time">
                <xs:sequence>
                        <xs:element name="Conference-occurrence" minOccurs="0"
maxOccurs="unbounded">
                                <xs:complexType>
```

```
                                        <xs:sequence>
                                                <xs:element name="Start-time"
type="xs:dateTime" minOccurs="0"/>
                                                <xs:element name="Stop-time"
type="xs:dateTime" minOccurs="0"/>
                                        </xs:sequence>
                                </xs:complexType>
                        </xs:element>
                </xs:sequence>
        </xs:complexType>
   </xs:schema>



   <!-- Access Control List ACL -->


   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-acl"
                        xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
        <xs:complexType name="Conference-ACL">
                <xs:sequence>
                        <xs:element name="ACL-target-URI"
maxOccurs="unbounded">
                                <xs:complexType>
                                        <xs:simpleContent>
                                                <xs:extension base="xs:anyURI">
                                                        <xs:attribute
name="Refer" type="xs:boolean" default="false"/>
                                                        <xs:attribute
name="Access-type" use="required">
                                                                <xs:simpleType>

<xs:restriction base="xs:string">

<xs:enumeration value="Allowed"/>

<xs:enumeration value="Blocked"/>

<xs:enumeration value="Pending"/>

<xs:enumeration value="Expelled"/>
                                                                        </
xs:restriction>
                                                                </
xs:simpleType>
                                                        </xs:attribute>
                                                </xs:extension>
                                        </xs:simpleContent>
                                </xs:complexType>
                        </xs:element>
                </xs:sequence>
        </xs:complexType>
   </xs:schema>
```

```
   <!-- Privilege Control List (PCL) -->
```

```
   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
               targetNamespace="urn:ietf:params:xml:ns:conference-pcl"
               elementFormDefault="qualified">
   <xs:complexType name="Conference-PCL">
   <xs:sequence>
        <xs:element name="PCL-target" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                <xs:sequence>
                        <xs:element name="PCL-target-uri" type="xs:anyURI"
minOccurs="1"/>
                        <xs:element name="Privileges">
                                <xs:simpleType>
                                <xs:list>
                                <!-- Define the privileges as data type with
all possible values -->
                                <xs:simpleType>
                                        <xs:restriction base="xs:string">
                                        <xs:enumeration
value="RIGHT_TO_SUBSCRIBE_TO_CONF_EVENT_PACKAGE"/>
                                        </xs:restriction>
                                </xs:simpleType>
                                </xs:list>
                                </xs:simpleType>
                        </xs:element>
                </xs:sequence>
                </xs:complexType>
        </xs:element>
   </xs:sequence>
   </xs:complexType>
   </xs:schema>
```

```
   <!-- Dial-Out List (DL) -->
```

```
   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-dl"
                        xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
        <xs:complexType name="Conference-DL">
                <xs:sequence>
                        <xs:element name="DL-target" maxOccurs="unbounded">
                                <xs:complexType>
                                        <xs:sequence>
                                                <xs:element name="DL-target-
URI" type="xs:anyURI"/>
                                        </xs:sequence>
                                </xs:complexType>
                        </xs:element>
                </xs:sequence>
        </xs:complexType>
   </xs:schema>



   <!-- Security Control (SC) -->


   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-sc"
xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
        <xs:complexType name="Conference-SC">
                <xs:sequence>
                        <xs:element name="Visibility">
                                <xs:simpleType>
                                        <xs:restriction base="xs:string">
                                                <xs:enumeration
value="visible"/>
                                                <xs:enumeration
value="invisible"/>
                                        </xs:restriction>
                                </xs:simpleType>
                        </xs:element>
                        <xs:element name="Security-mechanism">
                                <xs:complexType>
                                        <xs:attribute name="TLS"
type="xs:boolean" default="false"/>
                                        <xs:attribute name="S-MIME"
type="xs:boolean" default="false"/>
                                </xs:complexType>
                        </xs:element>
                        <xs:element name="SC-target" minOccurs="0"
maxOccurs="unbounded">
                                <xs:complexType>
                                        <xs:sequence>
                                                <xs:element name="SC-target-
URI" type="xs:anyURI"/>
                                                <xs:element
name="Authorization-mechanism">
                                                        <xs:simpleType>
                                                                <xs:restriction
base="xs:string">

<xs:enumeration value="Digest"/>

<xs:enumeration value="Digest-AKA"/>

<xs:enumeration value="None"/>
                                                                </
xs:restriction>
                                                        </xs:simpleType>
                                                </xs:element>
                                        </xs:sequence>
                                        <xs:attribute name="Password"
type="xs:string"/>
                                </xs:complexType>
                        </xs:element>
                </xs:sequence>
        </xs:complexType>
   </xs:schema>


   <!-- Floor policy  -->


   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-fp"
xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
        <xs:complexType name="Conference-floor-policy">
                <xs:sequence>
                        <xs:element name="Floor" maxOccurs="unbounded">
                                <xs:complexType>
                                        <xs:sequence>
                                                <xs:element name="Media-types">
```

```
                                                        <xs:complexType>
                                                                <xs:sequence>

<xs:element name="Video" minOccurs="0"/>

<xs:element name="Audio" minOccurs="0"/>

<xs:element name="Application" minOccurs="0"/>

<xs:element name="Data" minOccurs="0"/>

<xs:element name="Control" minOccurs="0"/>

<xs:element name="Message" minOccurs="0"/>

<xs:element name="Text" minOccurs="0"/>
                                                                        <xs:any
namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
                                                                </xs:sequence>
                                                        </xs:complexType>
                                                </xs:element>
                                                <xs:element name="Algorithm">
                                                        <xs:complexType>
                                                                <xs:sequence>

<xs:element name="Moderator-controlled" minOccurs="0"/>

<xs:element name="FCFS" minOccurs="0"/>

<xs:element name="Random" minOccurs="0"/>
                                                                        <xs:any
namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
                                                                </xs:sequence>
                                                        </xs:complexType>
                                                </xs:element>
                                                <xs:element name="Max-floor-
users" type="xs:nonNegativeInteger" minOccurs="0"/>
                                                <xs:element name="Moderator-
URI" type="xs:anyURI" minOccurs="0"/>
                                        </xs:sequence>
                                        <xs:attribute name="moderator-
controlled" type="xs:boolean" default="false"/>
                                </xs:complexType>
                        </xs:element>
                </xs:sequence>
        </xs:complexType>
   </xs:schema>
   <!-- Media policy-->


   <?xml version="1.0" encoding="UTF-8"?>
   <xs:schema targetNamespace="urn:ietf:params:xml:ns:conference-mp"
xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
        <xs:element name="Conference-Media-Policy">
                <xs:complexType>
                        <xs:sequence>
                                <xs:element name="Media-types">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element
name="Video" minOccurs="0"/>
                                                        <xs:element
name="Audio" minOccurs="0"/>
                                                        <xs:element
name="Application" minOccurs="0"/>
                                                        <xs:element name="Data"
minOccurs="0"/>
                                                        <xs:element
name="Control" minOccurs="0"/>
```

```
                                                        <xs:element
name="Message" minOccurs="0"/>
                                                        <xs:element name="Text"
minOccurs="0"/>
                                                        <xs:any
namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                        </xs:sequence>
                </xs:complexType>
        </xs:element>
   </xs:schema>
```

5. Floor Control Policy vs. Floor Control Protocol

Conference floor control is an optional feature provided by a separate floor control protocol (FCP). However, creating a floor and defining a floor policy belongs to CPCP. Moreover, setting some key floor parameters, such as floor moderator in moderator controlled floor policy, belongs to CPCP. FCP only defines how to request, grant, deny and revoke a floor within given floor policy.

For example, in a typical conference the privileged conference user (creator) uses CPCP for creating a floor for audio plane, defining the floor policy as "moderator-controlled" and appointing one user - possibly himself - to act as a floor moderator governing the access to the floor.

When the floor has been created and possible floor moderator has been assigned, the floor moderator gets notifications from the focus and is able to accept or deny floor requests from the conference users. Note that FCP does not create media streams (just the virtual floor attached to media), as media streams are created using CPCP. The details of FCP are beyond the scope of this draft.

6. An XCAP Usage for Conference Policy Manipulation

6.1 Application Unique ID

XCAP requires application usages to define a unique application usage ID (AUID) in either the IETF tree or a vendor tree. This specification defines the "conference-policy" AUID within the IETF tree, via the IANA registration in Section 9.

6.2 Resource Interdependencies

The conference policy server must fill the conference URI(s), if a conference URI was not proposed by the client. The client then needs to perform a HTTP GET to retrieve the modified policy containing the assigned conference URI(s). The CPS MAY assign multiple conference URIs to a conference, one for each call signaling protocol that it supports. Section 6.12 and Section 4.2.1 discuss this is more detail.

6.3 Additional Constraints

These are defined within the XML structure definition.

6.4 Naming Conventions

There are no naming conventions that need to be defined for this application usage.

6.5 Authorization Policies

This application usage does not modify the default XCAP authorization policy, which is that only a user can read, write or modify their own documents. A server can allow privileged users to modify documents that they don't own, but the establishment and indication of such policies is outside the scope of this document. It is anticipated that a future application usage will define which users are allowed to modify a list resource.

6.6 MIME Type for CPCP XML Document

The MIME type for the CPCP XML document is defined in Section 4.1

6.7 Overview of Operation

This document assumes that the user knows the location of conference policy server (the XCAP URI), the details of that discovery are beyond the scope of this document.

CPCP is implemented as an XCAP application usage [8].

CPCP allows clients to manipulate the conference policy at conference policy server (CPS). CPS is able to inform the focus about changes in conference policy, if necessary. For example, if new users are added to the dial-out list, then conference policy server informs the focus which makes the invitations as requested.

Some assumptions about the conferencing architecture are made. Clients always connect to the conference policy server (CPS) when they perform XCAP operations. It is assumed that CPS informs other conferencing entities, such as focus, floor control server and mixer directly or via focus. For example, if user A wants to expel user B from an ongoing conference, user A must first manipulate the conference policy data. CPS then communicates that change to the focus to perform the operation.

6.8 Communication Between Conference Entities

The communication between different (logical) conferencing elements is beyond the scope of this document. It can be expected that in most cases CPS includes also those logical functions. If the focus is not co-located with CPS, one way for the CPS to communicate changes to the conference policy is for the focus to subscribe to the XCAP event package [10].

6.9 Conference Creation and Termination

Conference is identified by one or more conference URIs. Conference URI assignment is discussed in Section 6.12 and Section 4.2.1.

A user may create a new conference at the CPS by using HTTP PUT and sending it to the CPS XCAP URI. Depending on server policy and user privileges, the CPS may accept the creation.

A conference can be deleted permanently using HTTP DELETE, which consequently frees the resources. When the user deletes a conference, CPS MUST also delete all its sub-conferences ("side bars") at a server. Conference side bars are separate (independent) URIs at the server.

A running conference instance can be also stopped by modifying the conference time information. This leaves conference ACLs and privileges intact but stops the conference.

If a conference is in progress when deleted or stopped, the focus issues signalling requests to terminate all conference related sessions it has with clients. In SIP, the focus issues BYE requests.

6.10 Manipulating the Participant Lists

A user with sufficient privileges is allowed to perform user management operations, such as adding a new user to the conference or expelling a user from the conference. These operations are performed by modifying the conference policy at the conference policy server. After authorising the user to do such manipulations, the conference policy server communicates the change to the focus. The focus reacts by performing operations such as sending SIP INVITE, BYE or REFER.

Asking the focus to invite a user into the conference is achieved by sending a HTTP PUT request to the CPS that modifies the Dial-Out List

(DL) adding URIs to it. The CPS then triggers the focus to send the conference invitation, eg: SIP INVITE(s) as needed. Similarly, a user can be removed from the Dial-out list by issuing a HTTP DELETE removing the URIs.

Asking the focus to allow certain users to join the conference is done by sending a HTTP PUT request to the CPS that modifies the ACL by adding URIs with access type of "Allowed". The CPS then informs the focus of such change to the ACL.

If the conference is long-lasting, it is possible that new rules are added all the time but old rules are almost never removed (some of them are overwritten, though). This leads easily to the situation that the ACL contains many unnecessary rules which are not really needed anymore. Therefore, there is a need to delete ACL rule. This can be achieved with the HTTP DELETE.

Conflicting rules MUST NOT exist (e.g. both allowed and blocked action is defined for same target). It is the responsibly of the CPS to ensure such restriction. If a conflict occurs, the CPS can ...

6.10.1 Expelling a Participant

Expel operation uses the HTTP PUT request as well, as the user is put on the ACL list with an access type of "Expelled". This also triggers the CPS to inform the focus about the need to issue a terminating request, such as a SIP BYE.

A participant cannot be expelled by placing him in the ACL list with an action to block. This is because the focus interprets a user placed on the block list as a user who is not allowed to dial into the conference, but does not prohibit the focus from inviting that user to join, if that user is on the Dial-out list. Having the user on an Expel list explicitly informs the focus not to invite that user, even if s/he is on the Dial-out list.

6.11 Privileges: Who can modify the conference policy

There is a need for different privileges to exist where users can modify certain parts of the conference policy XML document. This specification does not specify such privileges and relies on other XCAP usage documents to define those privileges. If no such XCAP usage document exists, the base XCAP document defines the default privileges so that only the creator of the document is the sole user with write access.

This specification, however, makes ready the CPCP XML document to allow an external usage document to define which parts of such an XML document a user can modify (which parts of an XML document a user has read/write access to) by dividing the CPCP XML document into sections, each with a separate namespace. It is envisioned that the XCAP usage document for read/write access of another XCAP XML document uses namespaces as the key to allow/disallow users from reading and/or modifying that XCAP usage document.

6.12 Conference URI(s)

A conference is identified by one or more conference URIs. Conference URIs can be proposed by the creator of the conference policy, as it may be useful to have human-friendly name in some cases, or can be assigned by the CPS. If the creator has proposed a conference URI, the server needs to decide whether it accept the name proposed by the client or not. It does this determination by examining if the conference URI already exists or not. If it exists, the server ...

A Conference URI can be SIP, SIPS, TEL, or any supported URI scheme. There must be at least one URI for a conference. The CPS MAY assign multiple conference URIs to a conference, one for each call signaling protocol that it supports. If the creator of the conference policy proposed a conference URI for a protocol that the server does not support, the server ...

7. Examples

The following is an example of a document compliant to the schema:

Below is an example how to create a conference:

1. Creating a Conference

Alice creates a conference as follows:

```
      PUT http://xcap.example.com/services/conferences/users/Alice/
conference.xml HTTP/1.1
      Content-Type:application/conference-policy+xml
```

```
   <?xml version="1.0" encoding="US-ASCII"?>
   <Conference xmlns="urn:ietf:params:xml:ns:conference-policy"
                        xmlns:conference-mp="urn:ietf:params:xml:ns:conference-
mp"
                        xmlns:conference-fp="urn:ietf:params:xml:ns:conference-
fp"
                        xmlns:conference-sc="urn:ietf:params:xml:ns:conference-
sc"
                        xmlns:conference-dl="urn:ietf:params:xml:ns:conference-
dl"
                        xmlns:conference-
pcl="urn:ietf:params:xml:ns:conference-pcl"
                        xmlns:conference-
acl="urn:ietf:params:xml:ns:conference-acl"
                        xmlns:conference-
time="urn:ietf:params:xml:ns:conference-time"
                        xmlns:conference-
info="urn:ietf:params:xml:ns:conference-info"
```

```
                        xmlns:conference-
settings="urn:ietf:params:xml:ns:conference-settings">
        <conference-settings:Conference-settings>
                <conference-uri:Conference-URI></conference-uri:Conference-URI>
                <Max-participant-count>50</Max-participant-count>
        </conference-settings:Conference-settings>
        <conference-info:Conference-info lang="en">
                <Subject>What's happening tonight</Subject>
                <Display-name>Party Goer's</Display-name>
                <Free-text>John and Peter will join the conference soon</Free-
text>
                <Keywords>party nightclub beer</Keywords>
                <Host-info>
                        <SIP-URI>sip:Alice@example.com</SIP-URI>
                        <TEL-URI>tel:+358401111111</TEL-URI>
                        <E-mail>mailto:Alice@example.com</E-mail>
                        <Web-page>http://www.example.com/users/Alice</Web-page>
                </Host-info>
        </conference-info:Conference-info>
        <conference-time:Conference-time>
                <Conference-occurrence>
                        <Start-time>2003-06-16T10:00:00Z</Start-time>
                        <Stop-time>2003-06-16T12:00:00Z</Stop-time>
                </Conference-occurrence>
        </conference-time:Conference-time>
        <conference-acl:ACL>
                <ACL-target-URI Access-type="Allowed">sip:*@example.com</ACL-
target-URI>
                <ACL-target-URI Access-type="Blocked">sip:*@*</ACL-target-URI>
        </conference-acl:ACL>
        <conference-pcl:PCL>
                <PCL-target>
                        <PCL-target-URI>sip:Alice@example.com</PCL-target-URI>
                        <Privileges>RIGHT_TO_SUBSCRIBE_TO_CONF_EVENT_PACKAGE</
Privileges>
                </PCL-target>
        </conference-pcl:PCL>
        <conference-dl:DL>
                <DL-target>
                        <DL-target-URI>sip:alice@operator.com</DL-target-URI>
                </DL-target>
                <DL-target>
                        <DL-target-URI>sip:sarah@operator.com</DL-target-URI>
                </DL-target>
        </conference-dl:DL>
        <conference-sc:SC>
                <Visibility>visible</Visibility>
                <Security-mechanism TLS="false" S-MIME="true"/>
                <SC-target>
                        <SC-target-URI>sip:*@example.com</SC-target-URI>
                        <Authorization-mechanism password="1a2b3c4d">Digest</
Authorization-mechanism>
                </SC-target>
```

```
        </conference-sc:SC>
        <conference-fp:Conference-floor-policy>
                <Floor moderator-controlled="true">
                        <Media-types>
                                <Audio/>
                        </Media-types>
                        <Algorithm>
                                <Moderator-controlled/>
                        </Algorithm>
                        <Max-floor-users>1</Max-floor-users>
                        <Moderator-URI>sip:Alice@example.com</Moderator-URI>
                </Floor>
        </conference-fp:Conference-floor-policy>
        <conference-mp:Conference-media-policy>
                        <Media-types>
                                <Audio/>
                        </Media-types>
        </conference-mp:Conference-media-policy>
   </Conference>
```

At exactly 2003-06-16T10:00:00Z, the conference server creates a focus and sends SIP INVITE requests to Alice and Sarah. After the focus is created, SIP INVITE requests can be accepted from anyone at domain example.com. Any attempts to join the conference by users in other domains are rejected.

2. Expelling a User

Continuing with the above example: aftar the conference has started, Alice decides to expel Bob who has joined the conference. So she adds him to the ACL list with Access-type of value "Blocked".

The XCAP request looks like:

```
      PUT http://xcap.example.com/services/conferences/users/Alice/
conference.xml?
         Conference/ACL/ACL-target-URI HTTP/1.1
         Content-Type:text/plain


      <ACL-target-URI Access-type="Explelled">sip:bob@example.com</ACL-target-
URI>
```

At this point, the focus sends a SIP BYE request to Bob ending Bob's participation in the conference. This also guarantees that Bob cannot rejoin the conference since he is expilictly expelled until his URI is removed from the ACL Expelled list. Any attempt Bob makes in rejoining the conference will fail.

3. Allowing An Expelled Participant To Join Again

Continuing with the example above, Alice now decides to allow Bob to join again after a period of time. She does so by removing his entry in the ACL that identifies him as "Expelled".

```
      DELETE http://xcap.example.com/services/conferences/users/Alice/
conference.xml?
         Conference/ACL/ACL-target-URI/ACL-target-URI="sip:bob@example.com"
HTTP/1.1
```

Bob can now rejoin the conference by sending a SIP INVITE request.

4. Removing A Conference

Alice now decides she no longer wants this conference to exist and therefore deletes the conference:

```
      DELETE http://xcap.example.com/services/conferences/users/Alice/
conference.xml
```

As a result of this action, the focus sends SIP BYE requests to all current participants in the conference. The conference server terminates the focus thereafter.

8. Security Considerations

See section Section 4.2.7.

9. IANA Considerations

9.1 XCAP Application Usage ID

This section registers a new XCAP Application Usage ID (AUID) according to the IANA procedures defined in..

Name of the AUID: conference-policy
Description: Conference policy application manipulates conference policy at a server.

9.2 application/conference-policy+xml mime TYPE

MIME media type: application

MIME subtype name: conference-policy+xml

Mandatory parameters: none

Optional parameters: Same as charset parameter applicatioN/xml as specified in RFC 3023 [6].

Encoding considerations: Same as encoding considerations of application/xml as specified in RFC 3023 [6].

Security considerations: See section 10 of RFC 3023 [6] and section Section 9 of this document.

Interoperability considerations: none.

Published specification: This document.

Applications which use this media type: This document type has been used to support conference policy manipulation for SIP based conferencing.

Additional information:

Magic number: None

File extension: .cl or .xml

Macintosh file type code: "TEXT"

Personal and email address for further information: Petri Koskelainen (petri.koskelainen@nokia.com)

Intended Usage: COMMON

Author/change controller: The IETF

9.3 URN Sub-Namespace Registration for urn:ietf:params:xml:ns:conference-policy

This section registers a new XML namespace, as per guidelines in URN document [13].

URI: The URI for this namespace is
urn:ietf:params:xml:ns:conference-policy.

Registrant Contact: IETF, XCON working group, Petri Koskelainen
(petri.koskelainen@nokia.com)

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
     "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
 <meta http-equiv="content-type"
   content="text/html;charset=iso-8859-1"/>
 <title>Conference Policy Namespace</title>
</head>
<body>
  <h1>Namespace for Conference Policy</h1>
  <h2>application/conference-policy+xml</h2>
  <p>See <a href="[[[URL of published RFC]]]">RFCXXXX</a>.</p>
</body>
</html>
END
```

10. Contributors

Jose Costa-Requena

Simo Veikkolainen

Teemu Jalava

11. Acknowledgements

The authors would like to thank Markus Isomaki, Eunsook Kim and IETF conferencing design team for their feedback.

Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, BCD 14, March 1997.

[2] Moats, R., "URN Syntax", RFC 2141, May 1997.

[3] Moats, R., "A URN Namespace for IETF Documents", RFC 2648, August 1999.

[4] Rosenberg et al., J., Shulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[5] Bray, T., Paoli, J., Sperberg-McQueen, C. and E. Maler,

"Extensible Markup Language (XML) 1.0 (Second Edition)", W3C REC REC-xml-20001006, October 2000.

[6] Murata, M., Laurent, S. and D. Kohn, "XML Media Types", RFC 3023, January 2001.

[7] Koskelainen, P. and H. Khartabil, "Requirements for conference policy control protocol", draft-ietf-xcon-cpcp-req-01 (work in progress), January 2004.

[8] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)", draft-ietf-simple-xcap-02 (work in progress), February 2004.

[9] Rosenberg, J., "An Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Usage for Presence Lists", draft-ietf-simple-xcap-list-usage-02 (work in progress), February 2004.

[10] Rosenberg, J., "A Session Initiation Protocol (SIP) Event Package for Modification Events for the Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Managed Documents", draft-ietf-simple-xcap-package-01 (work in progress), February 2004.

[11] Rosenberg, J., "A Framework for Conferencing with the Session Initiation Protocol", draft-ietf-sipping-conferencing-framework-01 (work in progress), October 2003.

[12] Rosenberg, J., Shulzrinne, H. and O. Levin, "A Session Initiation Protocol (SIP) Event Package for Conference State", draft-ietf-sipping-conference-package-03, February 2004.

[13] Mealling, M., "The IETF XML Registry", draft-mealling-iana-xmlns-registry-05 (work in progress), June 2003.

Authors' Addresses

Hisham Khartabil
Nokia
P.O. Box 321
Helsinki FIN-00045
Finland

EMail: hisham.khartabil@nokia.com

Petri Koskelainen
Nokia
P.O. Box 100 (Visiokatu 1)
Tampere FIN-33721
Finland

EMail: petri.koskelainen@nokia.com

Intellectual Property Statement

Full Copyright Statement

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.