3GPP TS 24.623 V11.1.0 (2012-12)

Technical Specification

3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Extensible Markup Language (XML) Configuration Access Protocol (XCAP) over the Ut interface for Manipulating Supplementary Services (Release 11)

Keywords
GSM, UMTS, LTE, access, CAP, configuration, service, supplementary service

3GPP

Postal address

3GPP support office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet
http://www.3gpp.org

Contents

Foreword

This Technical Specification (TS) was been produced by ETSI Technical Committee Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN) and originally published as ETSI TS 183 023 [13]. It was transferred to the 3rd Generation Partnership Project (3GPP) in January 2008.

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document defines a protocol used for manipulating data related to supplementary services. The protocol is based on the eXtensible Markup Language (XML) Configuration Access Protocol (XCAP) RFC 4825 [8]. A new XCAP application usage is defined for the purpose of manipulating the supplementary services data. The common XCAP related aspects that are applicable to supplementary services are specified in the present document. The protocol allows authorized users to manipulate service-related data either when they are connected to IMS or when they are connected to non-IMS networks (e.g. the public Internet).

The present document is applicable to User Equipment (UE) and Application Servers (AS) which are intended to support XCAP application usage for manipulating data related to supplementary services.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] IETF RFC 2616: "Hypertext Transfer Protocol - HTTP/1.1".

[2] W3C REC-xmlschema-1-20010502: "XML Schema Part 1: Structures".

[3] IETF RFC 2617: "HTTP Authentication: Basic and Digest Access Authentication".

[4] IETF RFC 2246: "The TLS Protocol Version 1.0".

[5] 3GPP TS 24.109: "Bootstrapping interface (Ub) and Network application function interface (Ua); Protocol details".

[6] 3GPP TS 33.222: "Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)".

[7] Void.

[8] IETF RFC 4825: "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)".

[9] Void

[10] Void.

[11] IETF RFC 5875 (May 2010): "An Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Diff Event Package".

[12] ETSI TS 183 038: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); PSTN/ISDN Simulation Services; Extensible Markup Language (XML) Document Management; Protocol Specification (Endorsement of OMA-TS-XDM-Core-V1-0-20051103-C and OMA-TS-XDM-Shared-V1-0-20051006-C)".

[13] ETSI TS 183 023 V1.4.0: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); PSTN/ISDN simulation services; Extensible Markup Language (XML) Configuration Access Protocol (XCAP) over the Ut interface for Manipulating NGN PSTN/ISDN Simulation Services".

[14] OMA-TS-XDM_Core-V1_1-20080627-A: "XML Document Management (XDM) Specification".

[15] 3GPP TS 23.003: "Numbering, addressing and identification".

[16] 3GPP TS 24.315: "IP Multimedia Subsystem (IMS) Operator Determined Barring (ODB); Stage 3".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in IETF RFC 4825 [8] apply.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AP	Authentication Proxy
AS	Application Server
AUID	Application Unique ID
HTTP	HyperText Transfer Protocol
ISDN	Integrated Services Digital Network
MIME	Multipurpose Internet Mail Extensions
NAF	Network Application Function
NGN	Next Generation Network
ODB	Operator Determined Barring
PSTN	Public Switched Telephone Network
TLS	Transport Layer Security
UE	User Equipment
URI	Uniform Resource Identifier
XCAP	XML Configuration Access Protocol
XML	eXtended Markup Language

4 Architecture for manipulating supplementary services settings

The protocol described in the present document allows to manipulate settings and variables related that influence the execution of one or more supplementary services. Manipulation of the supplementary services take place over the Ut interface (UE to AS), as shown in figure 1.

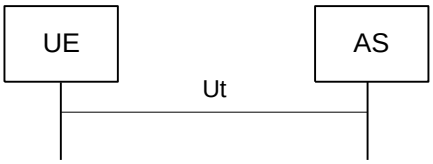

Figure 1: Ut interface

Manipulation of supplementary services does not usually take place during real-time operation. Typically users manipulate their services configuration data prior to the invocation and execution of the service.

Authentication of the user with HTTP may take place directly at the AS, such as in figure 1, or with the support of an Authentication Proxy, such as in figure 2. The architecture for authentication is provided in 3GPP TS 33.222 [6].

NOTE: The Network Application Function (NAF) can be an AS.

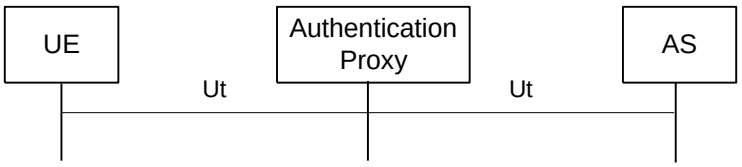

Figure 2: Authentication proxy in the Ut interface path

5 The eXtensible Markup Language (XML) Configuration Access Protocol (XCAP)

5.1 Introduction

For the purpose of manipulating data stored in an application server the XML Configuration Access Protocol (XCAP) [8] is used. XCAP allows a client to read, write and modify application configuration data, stored in XML format on a server. XCAP maps XML document sub-trees and element attributes to HTTP URIs, so that these components can be directly accessed by HTTP (see IETF RFC 2616 [1]). XCAP uses the HTTP methods PUT, GET, and DELETE to operating on XML documents stored in the server.

In the case of supplementary services, the data stored in a server is related to the execution of that given service. The present document defines a new XCAP Application Usage for the purpose of allowing a client to manipulate data related to supplementary services.

XCAP (see IETF RFC 4825 [8]) defines two logical roles: XCAP client and XCAP servers. An XCAP client is an HTTP/1.1 compliant client. Similarly an XCAP server is an HTTP/1.1 compliant server. The XCAP server acts as a repository of XML documents that customize and modify the execution of the supplementary services. Figure 3 depicts the XCAP architecture where an XCAP client sends an HTTP/1.1 request to an XCAP server. The server replies with an HTTP/1.1 response.

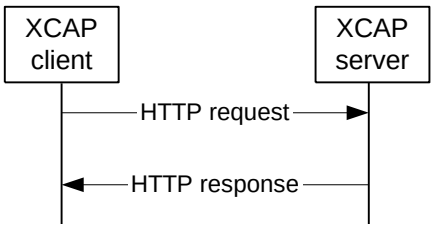

Figure 3: XCAP architecture

According to XCAP (see IETF RFC 4825 [8]), each application that makes use of XCAP defines its own XCAP application usage. The present document defines an supplementary services XCAP application usage in clause 6. This application usage defines the XML schema W3C REC-xmlschema-1-20010502 [2] for the data used by the application, along with other key pieces of information.

XCAP focuses on the definition of XML documents that are compliant with the XML schema and constrains defined for a particular XCAP application usage. XCAP allows application to provide XML documents that are common for all users or XML documents that affect the service of a given user.

Central to XCAP is the construction of the HTTP URI that points to particular XML document or certain components of it. A component in an XML document can be an XML element, attribute, or the value of it.

5.2 Functional entities

5.2.1 User Equipment (UE)

5.2.1.1 General

The UE implements the role of an XCAP client, as described in subclause 5.3.1 accessing the XCAP application usage as described in subclause 6.2.

For systems where Generic Authentication Architecture [6] is used, the UE shall support the authentication mechanisms specified in 3GPP TS 33.222 [6] and 3GPP TS 24.109 [5].

For systems where Generic Authentication Architecture [6] is not used, the UE shall support RFC 2617 [3] and IETF RFC 2246 [4] according to ETSI TS 183 038 [12].

On sending an HTTP request, the UE may indicate the user's identity intended to be used with the AS by adding a HTTP X-3GPP-Intended-Identity header (3GPP TS 24.109 [5]) to the outgoing HTTP request.

5.2.1.2 Subscription for notification of state changes in XML document

In order to keep the supplementary services state synchronized with the network elements and other terminals that the user might be using, the UE should subscribe to changes in the XCAP simserv documents by generating a SUBSCRIBE request in accordance with RFC 5875 [11].

5.2.2 Authentication Proxy (AP)

5.2.2.1 Introduction

An Authentication Proxy is an HTTP/1.1 IETF RFC 2616 [1] compliant server whose main purpose is to authenticate the user requests. The Authentication Proxy is used to separate the authentication procedure and the Application Server (AS) specific application logic to different logical entities.

The AP is configured as a HTTP reverse proxy, i.e. the FQDN of the AS is configured to the AP such a way that the IP traffic intended to the AS is directed to the AP by the network. The AP performs the authentication of the UE. After the authentication procedure has been successfully completed, the AP assumes the typical role of a reverse proxy, i.e. the AP forwards HTTP requests originating from the UE to the correct AS, and returns the corresponding HTTP responses from the AS to the originating UE.

The AP allows authorized users to manipulate services when they are connected to an IMS network or when they are connected to a non-IMS network (e.g. the public Internet). Authentication details can differ in both situations. Provisioning of credentials to authenticate the user is outside the scope of the present document. 3GPP TS 33.222 [6] provides further architectural authentication details.

5.2.2.2 Authentication

5.2.2.2.0 General

On receiving an HTTP request, the AP shall first determine the mechanism used to authenticate the user. If the Generic Authentication Architecture [6] is used, the AP shall attempt to authenticate the user via the mechanisms specified in 3GPP TS 33.222 [6] and the AP shall follow the procedures indicated in subclause 5.2.2.2.1. For systems where Generic Authentication Architecture 3GPP TS 33.222 [6] is not used, the AP shall attempt to authenticate the user according to IETF RFC 2617 [3] and ETSI TS 183 038 [12] provides guidelines for the Authentication Proxy.

5.2.2.2.1 Authentication based on the generic authentication architecture

On receiving an HTTP request that contains the Authorization header field, the AP shall:

a) use the value of that username parameter of the Authorization header field to authenticate the user;

b) apply the procedures specified in IETF RFC 2617 [3] for authentication;

c) if the HTTP request contains an X-3GPP-Intended-Identity header field (3GPP TS 24.109 [5]), then the AP may verify that the user identity belongs to the subscriber. This verification of the user identity shall be performed dependant on the subscriber's application specific or AP specific user security settings;

d) if authentication is successful, remove the Authorization header field from the HTTP request;

e) insert an HTTP X-3GPP-Asserted-Identity header field (3GPP TS 24.109 [5]) that contains the asserted identity or a list of identities; and

f) forward the HTTP request to the appropriate AS.

On receiving an HTTP response for the previous request, the AP shall:

a) add an Authentication-Info header field in accordance to the procedures described in 3GPP TS 33.222 [6]; and

b) forward the response to the XCAP client.

On receiving an HTTP request that does not contain the Authorization header field, the AP shall:

a) challenge the user by generating a 401 Unauthorized response according to the procedures specified in 3GPP TS 33.222 [6] and IETF RFC 2617 [3]; and

b) forward the 401 Unauthorized response to the sender of the HTTP request.

5.2.2.2.2 Void

5.2.2.3 Authorization

The AP shall be able to decide whether particular subscriber, i.e. the UE, is authorized to access a particular AS. On doing so, the AP may use the User Security Settings specified in 3GPP TS 24.109 [5].

The AP may indicate an asserted identity or a list of identities to the AS by adding an HTTP X-3GPP-Asserted-Identity header field to the HTTP requests prior to forwarding the request to the AS. In case of multiple identities, they shall be separated by comma (,) and each identity shall be surrounded by quotation marks ("). Whether the AP supports this handling of an asserted identity or a list of identities then it shall depend on local policy in the AP. In addition the subscriber's application specific or AP specific user security settings may be considered.

The AP may indicate an authorization flag or a list of authorization flags from the application specific user security settings (USS) to the AS by adding a HTTP X-3GPP-Authorization-Flags header field to the HTTP request prior to forward it to the XCAP server. The HTTP X-3GPP-Authorization-Flags header field shall contain a list of authorization flags separated by comma (,) and each authorization flag is surrounded by quotation marks ("). In case the AP supports this handling of authorization flags from USS then it shall depend on local policy in the AP.

5.2.3 Application Server (AS)

5.2.3.1 General

An Application Server implements the role of an XCAP server as described in subclause 5.3.2 providing the XCAP application usage as described in subclause 6.2.

For systems where Generic Authentication Architecture [6] is used, the AS shall support the authentication mechanisms specified in 3GPP TS 33.222 [6] and 3GPP TS 24.109 [5].

For systems where Generic Authentication Architecture [6] is not used, the AS shall support IETF RFC 2617 [3] and IETF RFC 2246 [4] according to ETSI TS 183 038 [12].

Procedures regarding Operator Determined Barring (ODB) are defined in 3GPP TS 24.315 [16].

5.2.3.2 Authentication and authorization

5.2.3.2.0 General

If an Authentication Proxy (AP) is provided in the path of the HTTP request, then the AS receives an HTTP request from a trusted source (the AP) and contains an HTTP X-3GPP-Asserted-Identity header (3GPP TS 24.109 [5]) that includes an asserted identity of the user. In this case the AS does not need to authenticate the user, but just provide authorization to access the requested resource.

If an HTTP X-3GPP-Asserted-Identity header (3GPP TS 24.109 [5]) is not present in the HTTP request or if the request is received from a non-trusted source, then the AS needs to authenticate the user prior to providing authorization to the XCAP resource by applying the procedures of authentication mechanisms specified in 3GPP TS 33.222 [6] and 3GPP TS 24.109 [5] in case Generic Authentication Architecture is supported, or as described in subclause 5.2.3.2.1 otherwise.

5.2.3.2.1 HTTP digest authentication

On receiving an HTTP request that does not contain an Authorization header the AS shall:

a) challenge the user by generating a 401 Unauthorized response that contains the proper Digest authentication parameters (e.g. realm), according to IETF RFC 2617 [3]. Provisioning of credentials to authenticate the user is outside the scope of the present document; and

b) forward the 401 Unauthorized response to the sender of the HTTP request.

On receiving an HTTP request that contains an Authorization header, the AS shall:

a) apply the authentication procedures defined in IETF RFC 2617 [3]; and

b) authorize or deny authorization depending on the authenticated identity.

5.2.3.3 Subscription acceptance and notification of state changes in XML document

When the AS receives a SUBSCRIBE request having the Event header field value set to "xcap-diff", the AS shall first authenticate the source of the SUBSCRIBE request and then perform authorization. Afterwards, the AS shall generate a response to the SUBSCRIBE request and notifications in accordance with RFC 5875 [11].

5.2.3.4 Validation against service capability

On receiving a XCAP request to modify service settings for a supplementary service, the AS shall check whether service capability fragments within the simservs document for the subscription of the sender for the XCAP request are available. If a service capability fragment within the simservs document is available, the AS shall validate the XCAP request against constraints defined in that service capability fragment and only accept modifications that are allowed by the service capability fragment. If the validation fails, the AS shall respond with a HTTP 409 (Conflict) response as defined in IETF RFC 4825 [8].

NOTE: The XML schema for a service capability fragment for a supplementary service is defined in the respective supplementary service specification.

5.3 Roles

5.3.1 XCAP client

5.3.1.1 Introduction

The XCAP client is a logical function as defined in IETF RFC 4825 [8]. The XCAP client provides the means to manipulate the general data, such as configuration settings related to supplementary services.

In order to manipulate XCAP resources stored on the XCAP server, the XCAP client uses the XCAP Root URI as defined in subclause 13.9.1 of 3GPP TS 23.003 [15]. The UE implementing the XCAP client can be provisioned with an XCAP Root URI as specified in Appendix C in OMA-TS-XDM_Core-V1_1-20080627-A [14].

NOTE: In order to be able to manipulate XCAP resources stored on the XCAP server, the XCAP client needs to know the user's directory name. It is assumed that this value is pre-provisioned or the UE uses some means to discover it. Discovery mechanisms are outside the scope of the present document.

5.3.1.2 Manipulating supplementary services

When the XCAP client intends to manipulate a resource list, it shall generate an HTTP PUT, HTTP GET or HTTP DELETE request in accordance with IETF RFC 4825 [8] and the supplementary services application usage specified in clause 6.

5.3.2 XCAP server

5.3.2.1 Introduction

The XCAP server is a logical function as defined in IETF RFC 4825 [8]. The XCAP server can store data related to the configuration of supplementary services. The XCAP server shall provide or deny authorization to access XCAP resources by authenticated users.

5.3.2.2 Manipulation acceptance

When the XCAP server receives an HTTP PUT, HTTP GET or HTTP DELETE request for manipulating or fetching a resource list, the XCAP server shall first authenticate the request and then perform authorization. Subclause 5.2.2 provides more details on the authentication and authorization of HTTP requests.

Afterwards the XCAP server shall perform the requested action and generate a response in accordance with IETF RFC 4825 [8] and the supplementary services application usage specified in clause 6.

6 Supplementary services XCAP application usage

6.1 Structure of the XML document

XCAP provides for the existence of application usages that define the conventions and constrains related to the manipulation of XML documents in an XCAP server. The present document defines a supplementary services XCAP application usage.

NOTE: Further releases can extend this application usage when deemed practical.

The present document follows a modular approach, as depicted in figure 4. We provide for the existence of a *simservs* XML document that contains the data associated to one or more supplementary services. The *simservs* XML document is composed of a common part, defined by the present document, and a number of XML fragments corresponding to each of the supplementary services. This modular approach has significant advantages. Particularly, it is versatile enough to allow any number of configurations. For example, in one configuration, an XCAP server might be managing a given server. In this case, the *simservs* XML document will contain one subtree per service. In another configuration, each service is managed in its own XCAP server, case in which the XML document in each XCAP server will contain the common parts and a single XML subtree that manages the service. Yet in a third configuration the XCAP server stores several XML documents, each document managing one or more services.

The XML schema for the *simservs* XML document, including the common parts, is specified in subclause 6.3. This XML schema allows for each of the individual XML schemas pertaining to a particular service to import the common parts XML schema. Each XML fragment affects the settings of a supplementary service (or group of services). The XML schema of each of the supplementary services is specified in its own specification. A template of the XML schema for a supplementary service is provided in subclause 6.4.

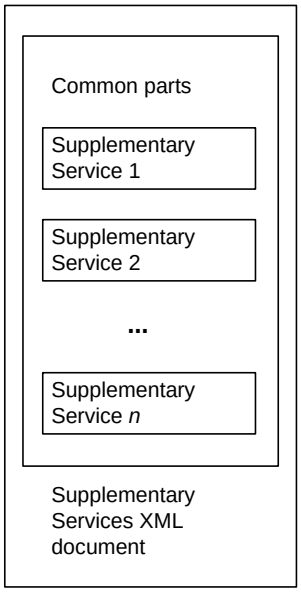

Figure 4: Structure of a supplementary services XML document

The *simservs* XML document starts with a <simservs> root XML element that can contain one or more child elements pertaining to supplementary services. Each of these service elements can contain an "active" attribute that indicates whether the service is activated or not. When the "active" attribute is absent on a service element, it indicates that the service is activated. Elements and attributes from different namespaces can be present as well. Services may also include capability elements that are read-only. These elements indicate which capabilities the network has provisioned for a user.

6.2 XCAP application usage

XCAP requires application usages to fulfil a number of steps in the definition of such application usage. The reminder of this clause specifies the required definitions of the supplementary services XCAP Application Usage.

Application Unique ID (AUID): Each XCAP application usage is associated with a unique name called the Application Unique ID (AUID). The AUID defined by this application usage falls into the vendor-proprietary namespace of XCAP AUID, where ETSI is considered a vendor.

The AUID allocated to the supplementary services XCAP application usage is:

`simservs.ngn.etsi.org`

XML schema: Implementations in compliance with the present document shall implement the XML schema that includes the XML Schema defined in clause 6.3. Additionally, each supplementary service (or group of them) is modelled with a XML fragment that is validated according to a specific XML schema. The XML schema that affects the settings of the related service is specified in the specification of the given supplementary service. Subclause 6.4 provides a template that shall be included in XML Schema that also includes the XML Schema defined in subclause 6.3 along with inclusion of XML schema defined by each of the supplementary services that implement XML schemas for data manipulation. Additionally the schema in subclause 6.3 contains the specification of a number of common service specific elements and types, the semantics and applicability of these elements is described in the service specifications that use them.

Default namespace: XCAP requires application usages to declare the default namespace. The default namespace of the supplementary services XCAP application usage is:

`http://uri.etsi.org/ngn/params/xml/simservs/xcap`

MIME type: The MIME type of supplementary services XML documents is:

`application/vnd.etsi.simservs+xml`

Validation constraints: The present document does not specify any additional constraint beyond those defined by XCAP RFC 4825 [8]. Note, however, that each of the supplementary services may specify additional constraints on each of the XML subdocuments.

Data semantics: The XML schema does not accept URIs that could be expressed as a relative URI reference causing a resolution problem. However, each of the supplementary services should consider if relative URIs are allowed in the subdocument tree, and in that case, they should indicate how to resolve relative URI references. In the absence of further indications, relative URI references should be resolved using the document URI as the base of the relative URI reference.

Naming conventions: By default, supplementary services XML documents are stored under the user's Home Directory (which is located under the "users" sub-tree). In order to facilitate the manipulation of a supplementary services XML document, we define a default XML file name:

`simservs.xml`

Resource interdependencies: The present document does not specify additional resource interdependency beyond those specified in the XML schema and beyond any resource interdependency that may be specified in each of supplementary services.

Authorization policies: The following authorization policy applies to the owner of *simservs* XML document:

a) authorised to retrieve any part of the document;

b) unauthorised to create:

   1) new child element(s) to the <simservs> root element; and

   2) new attribute(s) for a child element of the the <simservs> root element;

c) unauthorised to remove:

   1) existing child element(s) from the <simservs> root element; and

   2) existing attribute(s) from a child element of the <simservs> root element;

d) unauthorised to replace or remove:

   1) read-only child element(s) of the <simservs> root element, their attributes and their content;

e) unauthorised to replace:

   1) descendant element(s) of the <simservs> root element that are not allowed to be modified by the service capability fragments as described in subclause 5.2.3.4; and

   2) attribute(s) within descendant element(s) of the the <simservs> root element that are not allowed to be modified by the service capability fragments as described in subclause 5.2.3.4; and

f) authorized to replace element(s) and attribute(s) other than those specified in bullet d).

Users other than the owner of the *simservs* XML document are unauthorised to perform any operation on the document.

Unauthorized manipulation attempts on the *simservs* XML document are rejected with an HTTP 409 (Conflict) response as defined in IETF RFC 4825 [8].

NOTE 1: It is allowed to replace the *simservs* XML document or its <simservs> root element containing read-only child elements provided that the read-only child elements, including their content, are preserved.

NOTE 2: Any child elements of the <simservs> root element of the *simservs* XML document unknown to the XCAP client can be potentially read-only.

6.3 XML schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
              xmlns:ss="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
              xmlns:xs="http://www.w3.org/2001/XMLSchema"
              elementFormDefault="qualified"
              attributeFormDefault="unqualified">

<!-- The element "simservs" maps to the Common Parts of a supplementary services document -->

   <xs:element name="simservs">
      <xs:annotation>
         <xs:documentation>XML Schema for data manipulation of Supplementary
                           Services
         </xs:documentation>
      </xs:annotation>
      <xs:complexType>
         <xs:sequence>
            <xs:element ref="ss:absService" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element name="extensions" minOccurs="0">
               <xs:complexType>
                  <xs:sequence>
                     <xs:any namespace="##other" processContents="lax"
                        minOccurs="0" maxOccurs="unbounded"/>
                  </xs:sequence>
               </xs:complexType>
            </xs:element>
         </xs:sequence>
         <xs:anyAttribute namespace="##any" processContents="lax"/>
      </xs:complexType>
   </xs:element>

   <xs:element name="absService" abstract="true" type="ss:simservType"/>

   <xs:complexType name="simservType">
      <xs:attribute name="active" type="xs:boolean"
                    use="optional" default="true" />
      <xs:anyAttribute namespace="##any" processContents="lax"/>
   </xs:complexType>
   <xs:complexType name="provisioned-type">
      <xs:attribute name="provisioned" type="xs:boolean"
                    use="optional" default="true" />
      <xs:anyAttribute namespace="##any" processContents="lax"/>
   </xs:complexType>

   <xs:complexType name="supported-media-type">
      <xs:choice>
         <xs:element name="all-media" type="ss:empty-element-type"/>
         <xs:element name="no-media" type="ss:empty-element-type"/>
         <xs:sequence maxOccurs="unbounded">
            <xs:element name="media" type="ss:media-type"/>
         </xs:sequence>
         <xs:any namespace="##other" processContents="lax"/>
      </xs:choice>
   </xs:complexType>

   <xs:complexType name="provisioned-target-type">
      <xs:choice>
         <xs:element name="any-target-type" type="ss:empty-element-type"/>
         <xs:element name="telephony-type" type="ss:empty-element-type"/>
         <xs:any namespace="##other" processContents="lax"/>
      </xs:choice>
   </xs:complexType>

   <!-- service specific IETF common policy condition elements-->
   <xs:element name="anonymous" type="ss:empty-element-type"/>
   <xs:element name="presence-status" type="ss:presence-status-activity-type"/>
   <xs:element name="media" type="ss:media-type"/>
   <xs:element name="communication-diverted" type="ss:empty-element-type"/>
   <xs:element name="rule-deactivated" type="ss:empty-element-type"/>
   <xs:element name="not-registered" type="ss:empty-element-type"/>
   <xs:element name="busy" type="ss:empty-element-type"/>
   <xs:element name="no-answer" type="ss:empty-element-type"/>
   <xs:element name="not-reachable" type="ss:empty-element-type"/>
   <xs:element name="roaming" type="ss:empty-element-type"/>
   <xs:element name="international" type="ss:empty-element-type"/>
```

```
   <xs:element name="international-exHC" type="ss:empty-element-type"/>


   <!-- service specific type declarations -->
   <xs:simpleType name="media-type" final="list restriction">
      <xs:restriction base="xs:string"/>
   </xs:simpleType>
   <xs:simpleType name="presence-status-activity-type" final="list restriction">
      <xs:restriction base="xs:string"/>
   </xs:simpleType>
   <xs:complexType name="empty-element-type"/>

</xs:schema>
```

6.4 Template for a supplementary service XML schema

Supplementary services that implement XCAP operations to manipulate the data associated to its service shall base their XML schema in the following template. Replace "ServiceName" with the name or acronym of the actual service.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
   xmlns:ss="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
               elementFormDefault="qualified"
               attributeFormDefault="unqualified">

   <xs:element name="ServiceName" substitutionGroup="ss:absService">
      <xs:annotation>
         <xs:documentation>Template of a
                                 Supplementary Service XML Schema
         </xs:documentation>
      </xs:annotation>

      <!-- If the service needs to add children elements or attributes -->
      <!-- it can use the following complexType for such purpose -->
      <xs:complexType>
          <xs:complexContent>
             <xs:extension base="ss:simservType">
                   <xs:sequence>
                      <!-- service specific elements can be defined here -->
                   </xs:sequence>
                   <!-- service specific attributes can be defined here -->
             </xs:extension>
          </xs:complexContent>
      </xs:complexType>

   </xs:element>
</xs:schema>
```

Annex A (informative):
Void

Annex B (normative): Connectivity Aspects when using XCAP

B.1 Scope

The present annex defines aspects for the connection between UE and the network to be used for XCAP.

B.2 Procedures at the UE

In order to manipulate XCAP resources stored on the XCAP server, a UE can be configured with parameters describing a connection to be used for XCAP. Connection parameters can be configured as specified in appendix C in OMA-TS-XDM_Core-V1_1-20080627-A [14].

Annex C (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2005-09					Publication as **ETSI TS 183 023**		1.1.1
2006-03					Publication as ETSI TS 183 023		1.2.1
2007-04					Publication as ETSI TS 183 023		1.3.1
2008-01					Publication as ETSI TS 183 023		1.4.0
2008-01					Conversion to **3GPP TS 24.423**		1.4.1
2008-01					Technically identical copy as **3GPP TS 24.623** as basis for further development.		1.4.2
2008-02					CT1#51 agreed to sent spec for information to plenary		1.4.3
2008-04					The following CR's were incorporated and the editor adopted their content / structure to the structure of the TS C1-081006	1.4.3	1.5.0
2008-05					The following CR's were incorporated and the editor adopted their content / structure to the structure of the TS C1-081616 C1-081894 C1-081916	1.5.0	1.6.0
2008-05					Editorial changes done by MCC	1.6.0	1.6.1
2008-06	CT#40	CP-080333			CP-080333 was approved by CT#40 and version 8.0.0 is created by MCC for publishing	1.6.1	8.0.0
2008-06					Version 8.0.1 created to inlcude attachments (.xml adn .xsd files)	8.0.0	8.0.1
2008-09	CT#41	CP-080533	0001	1	Tidyup .xml and .xsd files	8.0.1	8.1.0
2008-09	CT#41	CP-080533	0002		Applicability statement in scope	8.0.1	8.1.0
2009-06	CT#44	CP-090432	0005	2	Addition of international-communications condition	8.1.0	9.0.0
2009-09	CT#45	CP-090687	0007	1	Validation against capabilities	9.0.0	9.1.0
2009-09	CT#45	CP-090687	0008	1	Supported-media-type	9.0.0	9.1.0
2009-09	CT#45	CP-090687	0010		Supported target type	9.0.0	9.1.0
2009-09	CT#45				Editorial cleanup by MCC	9.1.0	9.1.1
2009-12	CT#46	CP-091040	0012	1	Change of ua-profile package to xcap-diff package	9.1.1	9.2.0
2009-12	CT#46	CP-090928	0013	2	Authorization policy update	9.1.1	9.2.0
2009-12	CT#46	CP-090928	0014	1	Service capabilities fragment	9.1.1	9.2.0
2009-12	CT#46	CP-090928	0015		Correct .xml schema	9.1.1	9.2.0
2011-03	CT#51				Upgrade to Rel-10	9.2.0	10.0.0
2011-09	CT#53	CP-110657	0021		IETF reference updates	10.0.0	10.1.0
2011-12	CT#54	CP-110857	0024	1	Incorrect MIME definition	10.1.0	10.2.0
2012-03	CT#55	CP-120097	0027	1	Connection to be used for XCAP	10.2.0	10.3.0
2012-03	CT#55	CP-120097	0030	1	Configuration of XCAP root URI	10.2.0	10.3.0
2012-09	CT#57				Upgrade to Rel-11	10.3.0	11.0.0
2012-12	CT#58	CP-120816	0032	2	Reference to ODB specification for ut based service configuration	11.0.0	11.1.0