

SIMPLE  
Internet-Draft  
Expires: January 11, 2006

J. Rosenberg  
Cisco Systems  
July 10, 2005

An Extensible Markup Language (XML) Representation for Expressing Policy Capabilities  
[draft-ietf-simple-common-policy-caps-00](#)

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on January 11, 2006.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

An important component of presence and location services is policy. Policy systems allow the presentity or location target to grant access to specific pieces of information to specific watchers or requestors. These policy systems can be extremely simple, allowing a user to accept or block requests based solely on the identity of the requestor, to extremely complex, allowing for time based rules that grant or deny specific pieces of information. Policy systems often support vendor proprietary features. To allow for interoperability

Internet-Draft

Common Policy Capabilities

July 2005

between clients which set such policies, and servers which execute them, it is necessary for clients to be able to determine the capabilities of the server to which it is connected. This specification defines an Extensible Markup Language (XML) based format for expressing such capabilities.

## Table of Contents

<a href="#">1.</a>	<a href="#">Terminology . . . . .</a>	<a href="#">3</a>
<a href="#">2.</a>	<a href="#">Introduction . . . . .</a>	<a href="#">3</a>
<a href="#">3.</a>	<a href="#">Overview of Operation . . . . .</a>	<a href="#">3</a>
<a href="#">4.</a>	<a href="#">Structure of Policy Capabilities . . . . .</a>	<a href="#">4</a>
<a href="#">5.</a>	<a href="#">XML Schema . . . . .</a>	<a href="#">6</a>
<a href="#">6.</a>	<a href="#">Example Document . . . . .</a>	<a href="#">7</a>
<a href="#">7.</a>	<a href="#">Usage with XCAP . . . . .</a>	<a href="#">7</a>
<a href="#">7.1</a>	<a href="#">Application Unique ID . . . . .</a>	<a href="#">7</a>
<a href="#">7.2</a>	<a href="#">XML Schema . . . . .</a>	<a href="#">7</a>
<a href="#">7.3</a>	<a href="#">Default Namespace . . . . .</a>	<a href="#">8</a>
<a href="#">7.4</a>	<a href="#">MIME Type . . . . .</a>	<a href="#">8</a>
<a href="#">7.5</a>	<a href="#">Validation Constraints . . . . .</a>	<a href="#">8</a>
<a href="#">7.6</a>	<a href="#">Data Semantics . . . . .</a>	<a href="#">8</a>
<a href="#">7.7</a>	<a href="#">Naming Conventions . . . . .</a>	<a href="#">8</a>
<a href="#">7.8</a>	<a href="#">Resource Interdependencies . . . . .</a>	<a href="#">8</a>
<a href="#">7.9</a>	<a href="#">Authorization Policies . . . . .</a>	<a href="#">8</a>
<a href="#">8.</a>	<a href="#">Security Considerations . . . . .</a>	<a href="#">9</a>
<a href="#">9.</a>	<a href="#">IANA Considerations . . . . .</a>	<a href="#">9</a>
<a href="#">9.1</a>	<a href="#">XCAP Application Usage ID . . . . .</a>	<a href="#">9</a>
<a href="#">9.2</a>	<a href="#">MIME Type Registration . . . . .</a>	<a href="#">9</a>
<a href="#">9.3</a>	<a href="#">URN Sub-Namespace Registrations . . . . .</a>	<a href="#">10</a>
<a href="#">9.4</a>	<a href="#">XML Schema Registration . . . . .</a>	<a href="#">11</a>
<a href="#">10.</a>	<a href="#">References . . . . .</a>	<a href="#">11</a>
<a href="#">10.1</a>	<a href="#">Normative References . . . . .</a>	<a href="#">11</a>
<a href="#">10.2</a>	<a href="#">Informative References . . . . .</a>	<a href="#">12</a>
	<a href="#">Author's Address . . . . .</a>	<a href="#">13</a>
	<a href="#">Intellectual Property and Copyright Statements . . . . .</a>	<a href="#">14</a>

## 1. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC 2119](#) [[1](#)] and indicate requirement levels for compliant implementations.

## 2. Introduction

An important component of presence [[10](#)] and location services [[11](#)] is policy. Policy systems allow the presentity or location target (referred to generically as the Presentity Target (PT)) to grant access to specific pieces of information to specific watchers or requestors (referred to as a WR) [[5](#)]. These policy systems can be extremely simple, allowing a PT to accept or block requests based solely on the identity of the WR, to extremely complex, allowing for time based rules that grant or deny specific pieces of information. [[5](#)] specifies a generic format for representing these policies, using the Extensible Markup Language (XML). These policies consist of conditions, actions, and transformations. That specification defines very few actual conditions, actions or transformations. Rather, it leaves such definitions to actual policy systems, such as [[5](#)] for location services, and [[13](#)] for presence services.

In addition to the conditions, actions and transformations specified in the documents referenced above, policy systems often support vendor proprietary features. It is also anticipated that future specifications will be continually developed that add new types of policies. This presents an interoperability challenge. Clients may support policies that are not supported by the servers they are using. This could lead to protocol failures or poor user experiences.

To address this problem, it is necessary for a capability declaration system to be put in place. This specification defines a general purpose format for representing policy capabilities within the

framework established in [5].

### 3. Overview of Operation

This specification defines an XML-based document format that allows a server to represent its capabilities. When a client, acting as an agent of a PT, starts up, it obtains this document from its policy server. This specification does not prescribe a singular means of transporting such a document between the server and the client. It is anticipated that different systems may use different techniques. However, for systems that make use of the XML Configuration Access Protocol (XCAP) [4], Section 7 defines an application usage that

allows for the transfer of the document using XCAP.

Once the document has been obtained by the client, it can determine which actions, conditions and transformations are understood by the server. This set is matched against those supported by the client. Those actions, conditions and transformations supported by the client, but not by the server, can be "greyed out" from a user interface, for example.

It is anticipated that the capabilities of the server can change over time. As a result, it is RECOMMENDED that clients obtain a fresh copy of the capabilities document each time they start.

### 4. Structure of Policy Capabilities

A policy capabilities document is an XML [6] document that MUST be well-formed and SHOULD be valid. Policy capabilities documents MUST be based on XML 1.0 and MUST be encoded using UTF-8. This specification makes use of XML namespaces for identifying policy capabilities documents and document fragments. The namespace URI for elements defined for this purpose is a URN [2], using the namespace identifier 'ietf' defined by [3] and extended by [7]. This URN is:

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

A policy capabilities document is structured much like a policy document [5]. The root element is <policy-capabilities>. This element has three children - <conditions>, <actions>, and <transformations>. Each of these contain a list of the condition,

action, and transformation capabilities, respectively. Generally speaking, each specific condition, action or transformation element (referred to as a capability element) is empty, unless it requires additional content to further refine the capability.

This specification defines three capability elements - <identity>, <validity>, and <sphere> matching the three conditions defined in [5]. Other specifications that define additional policies MUST also define matching capability elements. When such elements are defined, that specification MUST indicate, for each capability element, the corresponding action, condition, or permission elements which can be placed into a common policy document [5]. The name of the capability element need not match the name of the corresponding condition, action or transformation, although using a matching name is RECOMMENDED.

A server constructing a document to represent its capabilities MUST include all of its capabilities, even if those capabilities represent mandatory-to-implement features. However, the server MAY indicate

differing sets of capabilities to different users. As such, the set of capabilities combines both the ability and the willingness to support those capabilities.

## [5.](#) XML Schema

```
<xs:schema
  targetNamespace="urn:ietf:params:xml:ns:policy-capabilities"
  xmlns="urn:ietf:params:xml:ns:policy-capabilities"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="policy-capabilities">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="conditions" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element ref="identity"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

        <xs:element ref="sphere"/>
        <xs:element ref="validity"/>
        <xs:any namespace="##other" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="actions" minOccurs="0">
    <xs:complexType>
        <xs:sequence>
            <xs:any namespace="##other" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="transformations" minOccurs="0">
    <xs:complexType>
        <xs:sequence>
            <xs:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:complexType name="emptyType">
    <xs:complexContent>
        <xs:restriction base="xs:anyType"/>
    </xs:complexContent>
</xs:complexType>
<xs:element name="validity" type="emptyType"/>
<xs:element name="sphere" type="emptyType"/>
<xs:element name="identity" type="emptyType"/>
</xs:schema>

```

## [6.](#) Example Document

The following document indicates that the identity, validity and sphere conditions are supported. It also indicates that a vendor-specific capability, called <happy>, is supported, a vendor specific action capability, <log>, is supported, and a vendor-specific transformation capability - <min-security> is supported. The specification which defines these capabilities would indicate the

specific elements which could be included in a policy document.

```
<?xml version="1.0" encoding="UTF-8"?>
<policy-capabilities
  xmlns="urn:ietf:params:xml:ns:policy-capabilities"
  xmlns:cp="urn:ietf:params:xml:ns:policy-capabilities"
  xmlns:vpp="urn:ietf:params:xml:ns:vpp"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <conditions>
    <identity/>
    <sphere/>
    <validity/>
    <vpp:happy/>
  </conditions>
  <actions>
    <vpp:log/>
  </actions>
  <transformations>
    <vpp:min-security/>
  </transformations>
</policy-capabilities>
```

## [7.](#) Usage with XCAP

The following section defines the details necessary for clients to read supported permissions documents from a server using XCAP.

### [7.1](#) Application Unique ID

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

### [7.2](#) XML Schema

The schema is defined in [Section 5](#).

### [7.3](#) Default Namespace



The default namespace used in evaluating a URI is `urn:ietf:params:xml:ns:policy-capabilities`.

#### [7.4](#) MIME Type

The MIME type for this document is `"application/policy-caps+xml"`.

#### [7.5](#) Validation Constraints

This specification does not introduce any additional validation constraints beyond those defined in the schema.

#### [7.6](#) Data Semantics

Semantics for the document content are provided in [Section 4](#).

#### [7.7](#) Naming Conventions

When a client starts, it can fetch the capabilities of the server in one of two places. If the server capabilities differ on a user by user basis, the capabilities for user foo can be found in the document with filename `"cap.xml"` in the user's home directory for this application usage. A client SHOULD check this file first. If this document doesn't exist, the client should next check for the system wide permissions by reading the document with filename `"cap.xml"` in the global directory for this application usage.

#### [7.8](#) Resource Interdependencies

Policy capability documents are usually either created automatically by the server, or modified by administrator to reflect the features of a server. For those users that have access to the full capabilities of the server, a change in the server-wide capabilities, expressed in the `"cap.xml"` file in the global directory, MUST be reflected in any `"cap.xml"` documents in user's home directories.

#### [7.9](#) Authorization Policies

This application usage does not use the default XCAP authorization policies.

A user cannot modify the supported permissions document, they can only read it. Write access is granted only to administrators.

A user can read the `"cap.xml"` document in the global directory, but cannot modify it. Write access is granted only to administrators.

## [8.](#) Security Considerations

Policy capability documents reveal capability information about a server. This information can potentially be used by an enterprise to determine the features found in competitive products. However, such information could just as easily be obtained through other means, for example, by signing up as a legitimate user of the competitive service. Because supported permission documents can vary by user to user, they can also reveal information about the grade of service offered to a particular user. However, this information does not appear particularly sensitive. As a result, encryption of these documents is not terribly important.

If an attacker can modify the contents of a supported permission document as it passes from client to server, the attacker can remove capability elements, therefore reducing the level of service received by the client. This can therefore form a type of denial-of-service attack. As a result, systems which transfer these documents SHOULD provide for message integrity.

## [9.](#) IANA Considerations

There are several IANA considerations associated with this specification.

### [9.1](#) XCAP Application Usage ID

This section registers an XCAP Application Unique ID (AUID) according to the IANA procedures defined in [\[4\]](#).

Name of the AUID: policy-capabilities

Description: Policy capability documents describe the capabilities of a policy server to support different conditions, actions, and transformations, as defined in [\[5\]](#).

### [9.2](#) MIME Type Registration

This specification requests the registration of a new MIME type according to the procedures of [RFC 2048](#) [\[8\]](#) and guidelines in [RFC 3023](#) [\[9\]](#).

MIME media type name: application

Internet-Draft

Common Policy Capabilities

July 2005

MIME subtype name: policy-caps+xml

Mandatory parameters: none

Optional parameters: Same as charset parameter application/xml as specified in [RFC 3023](#) [9].

Encoding considerations: Same as encoding considerations of application/xml as specified in [RFC 3023](#) [9].

Security considerations: See [Section 10 of RFC 3023](#) [9] and [Section 8](#) of RFC XXXX [[NOTE TO IANA/RFC-EDITOR: Please replace XXXX with the RFC number of this specification]].

Interoperability considerations: none.

Published specification: RFC XXXX [[NOTE TO IANA/RFC-EDITOR: Please replace XXXX with the RFC number of this specification]]

Applications which use this media type: This document type has been used to support capabilities for presence and geolocation.

Additional Information:

Magic Number: None

File Extension: .pcp

Macintosh file type code: "TEXT"

Personal and email address for further information: Jonathan Rosenberg, [jdrosen@jdrosen.net](mailto:jdrosen@jdrosen.net)

Intended usage: COMMON

Author/Change controller: The IETF.

### [9.3](#) URN Sub-Namespace Registrations

This section registers a new XML namespace, as per the guidelines in [\[7\]](#)

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

Rosenberg

Expires January 11, 2006

[Page 10]

---

Internet-Draft

Common Policy Capabilities

July 2005

Registrant Contact: IETF, SIMPLE working group, (simple@ietf.org),  
Jonathan Rosenberg (jdrosen@jdrosen.net).

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>Policy Capabilities Namespace</title>
</head>
<body>
  <h1>Namespace for Policy Capabilities</h1>
  <h2>urn:ietf:params:xml:ns:policy-capabilities</h2>
  <p>See <a href="[URL of published RFC]">RFCXXXX[[NOTE
TO IANA/RFC-EDITOR: Please replace XXXX with the RFC number for this
specification.</a>.</p>
</body>
</html>
END
```

#### [9.4](#) XML Schema Registration

This section registers an XML schema as per the procedures in [\[7\]](#).

URI: urn:ietf:params:xml:schema:policy-capabilities

Registrant Contact: IETF, SIMPLE working group, ([simple@ietf.org](mailto:simple@ietf.org)), Jonathan Rosenberg ([jdrosen@jdrosen.net](mailto:jdrosen@jdrosen.net)).

The XML for this schema can be found as the sole content of [Section 5](#).

## [10](#). References

### [10.1](#) Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [2] Moats, R., "URN Syntax", [RFC 2141](#), May 1997.

Rosenberg	Expires January 11, 2006	[Page 11]
-----------	--------------------------	-----------

---

Internet-Draft	Common Policy Capabilities	July 2005
----------------	----------------------------	-----------

- [3] Moats, R., "A URN Namespace for IETF Documents", [RFC 2648](#), August 1999.
- [4] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)", [draft-ietf-simple-xcap-07](#) (work in progress), June 2005.
- [5] Schulzrinne, H., "A Document Format for Expressing Privacy Preferences", [draft-ietf-geopriv-common-policy-04](#) (work in progress), February 2005.
- [6] Bray, T., Paoli, J., Sperberg-McQueen, C., and E. Maler, "Extensible Markup Language (XML) 1.0 (Second Edition)", W3C FirstEdition REC-xml-20001006, October 2000.
- [7] Mealling, M., "The IETF XML Registry", [BCP 81](#), [RFC 3688](#), January 2004.
- [8] Freed, N., Klensin, J., and J. Postel, "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", [BCP 13](#), [RFC 2048](#), November 1996.
- [9] Murata, M., St. Laurent, S., and D. Kohn, "XML Media Types", [RFC 3023](#), January 2001.

## 10.2 Informative References

- [10] Day, M., Rosenberg, J., and H. Sugano, "A Model for Presence and Instant Messaging", [RFC 2778](#), February 2000.
- [11] Cuellar, J., Morris, J., Mulligan, D., Peterson, J., and J. Polk, "Geopriv Requirements", [RFC 3693](#), February 2004.
- [12] Schulzrinne, H., "A Document Format for Expressing Privacy Preferences for Location Information", [draft-ietf-geopriv-policy-05](#) (work in progress), November 2004.
- [13] Rosenberg, J., "Presence Authorization Rules", [draft-ietf-simple-presence-rules-02](#) (work in progress), February 2005.

Rosenberg

Expires January 11, 2006

[Page 12]

---

Internet-Draft

Common Policy Capabilities

July 2005

### Author's Address

Jonathan Rosenberg  
Cisco Systems  
600 Lanidex Plaza  
Parsippany, NJ 07054  
US

Phone: +1 973 952-5000  
Email: [jdrosen@cisco.com](mailto:jdrosen@cisco.com)  
URI: <http://www.jdrosen.net>

#### Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

#### Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

#### Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

#### Acknowledgment

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