



# XMPP

## XEP-0104: HTTP Scheme for URL Data

Matthew Miller

<mailto:linuxwolf@outer-planes.net>

<xmpp:linuxwolf@outer-planes.net>

2004-01-20

Version 0.3

Status	Type	Short Name
Deferred	Standards Track	N/A

This document provides a schema description for detailed information about HTTP URLs.

# Legal

## Copyright

This XMPP Extension Protocol is copyright © 1999 - 2011 by the [XMPP Standards Foundation](#) (XSF).

## Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the "Specification"), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

## Warranty

## NOTE WELL: This Specification is provided on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. ##

## Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

## Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF's Intellectual Property Rights Policy (a copy of which can be found at <http://xmpp.org/about-xmpp/xsf/xsf-ipr-policy/> or obtained by writing to XMPP Standards Foundation, 1899 Wynkoop Street, Suite 600, Denver, CO 80202 USA).

# Contents

1	Introduction	1
2	Requirements	1
3	Basic Usage	1
3.1	Providing Authentication . . . . .	1
3.2	Providing Cookies . . . . .	2
3.3	Providing Headers . . . . .	3
4	Implementation Notes	3
4.1	Cookies . . . . .	3
4.2	Headers . . . . .	3
5	Formal Description	4
5.1	Overview . . . . .	4
5.2	<auth/> Element . . . . .	4
5.3	<auth-param/> Element . . . . .	4
5.4	<cookie/> Element . . . . .	4
5.5	<header/> Element . . . . .	5
5.6	XML Schema . . . . .	5
6	Security Considerations	6
6.1	Authentication Information . . . . .	6
7	IANA Considerations	7
8	XMPP Registrar Considerations	7

## 1 Introduction

The most common URL scheme distributed over the Internet is HTTP and HTTPS. This document defines a structure that extends [URL Address Information](#)<sup>1</sup> to enable more advanced access to such URLs within Jabber.

## 2 Requirements

This document supplements XEP-0103 to provide more detailed information about HTTP and HTTPS URLs. The requirements this document fulfills are:

- Provide authentication information.
- Provide cookie data.
- Provide necessary headers.

The intent of this information is to provide an HTTP client with enough information in order to construct the HTTP request and entity headers necessary, as defined in [RFC 2616](#)<sup>2</sup>.

The use of this document in conjunction with XEP-0103 is OPTIONAL. The entity sending the URL is not required to provide any of this information, and receiving entities MAY ignore it.

## 3 Basic Usage

The two most typical types of information that can be necessary for accessing an HTTP URL are authentication details and cookies. In some cases, custom headers MAY also be necessary for successful use. Authentication information is provided in a scheme-independent format. Cookie data provided includes what would be necessary for a client to properly persist the value.

### 3.1 Providing Authentication

At a minimum, this document allows for an entity to indicate what authentication scheme is in use:

Listing 1: Indicating auth scheme

```
<message to='client@domain.com'>  
  <url-data xmlns='http://jabber.org/protocol/url-data'
```

<sup>1</sup>XEP-0103: URL Address Information <<http://xmpp.org/extensions/xep-0103.html>>.

<sup>2</sup>RFC 2616: Hypertext Transport Protocol -- HTTP/1.1 <<http://tools.ietf.org/html/rfc2616>>.

```

    xmlns:http='http://jabber.org/protocol/url-data/scheme/http'
    target='http://www.jabber.org/members/index.php'>
  <http:auth scheme='basic' />
</url-data>
</message>

```

To provide additional parameters (such as a realm and username/password), the <auth-param/> element is used:

```

<message to='client@domain.com'>
  <url-data xmlns='http://jabber.org/protocol/url-data'
    xmlns:http='http://jabber.org/protocol/url-data/scheme/http'
    target='http://www.jabber.org/members/index.php'>
    <http:auth scheme='basic'>
      <http:auth-param name='realm' value='www.jabber.org' />
      <auth-param name='username' value='defaultuser' />
      <auth-param name='password' value='defaultpwd' />
    </http:auth>
  </url-data>
</message>

```

### 3.2 Providing Cookies

Cookie information is provided by the <cookie/> element. This element can provide all of the information of the "Set-Cookie" response header<sup>3</sup>. The simplest usage is:

```

<message to='client@domain.com'>
  <url-data xmlns='http://jabber.org/protocol/url-data'
    xmlns:http='http://jabber.org/protocol/url-data/scheme/http'
    target='http://www.jabber.org/members/index.php'>
    <http:cookie name='jsessionid' value='1243asd234190sa32ds' />
  </url-data>
</message>

```

The above cookie is considered "transient", and will terminate when the HTTP session ends. Additional information about the cookie can be provided:

```

<message to='client@domain.com'>
  <url-data xmlns='http://jabber.org/protocol/url-data'
    xmlns:http='http://jabber.org/protocol/url-data/scheme/http'
    target='http://www.jabber.org/members/index.php'>
    <http:cookie name='jsessionid'
      domain='jabber.org'

```

<sup>3</sup>"Set-Cookie" is a message header for the HTTP response, and the <header/> element represents only message headers for HTTP requests. Therefore, cookies are handled differently.

```

    max-age='1234000'
    path='/members'
    comment='Web_Session_Identifier'
    version='1.0'
    secure='false'
    value='1243asd234190sa32ds' />
</url-data>
</message>

```

As demonstrated, the `<cookie/>` provides all the attributes provided by the "Set-Cookie" header. The only attributes required are "name" and "value".

### 3.3 Providing Headers

In some cases, arbitrary headers are necessary. They are provided using the `<header/>` element:

```

<message to='client@domain.com'>
  <url-data xmlns='http://jabber.org/protocol/url-data'
    xmlns:http='http://jabber.org/protocol/url-data/scheme/http'
    target='http://www.jabber.org/members/index.php'>
    <http:header name='Custom-Data' value='some_custom_data' />
  </url-data>
</message>

```

## 4 Implementation Notes

### 4.1 Cookies

Entities receiving cookie information SHOULD following the state management guidelines from [RFC 2109](http://tools.ietf.org/html/rfc2109)<sup>4</sup>. In particular, the interpretation of the "persistence details" attribute values from `<cookie/>` is that defined in section 4.2.2 of RFC 1209.

The attributes provided by `<cookie/>` MUST NOT be "pre-escaped" for HTTP message headers. It is the responsibility of the receiving entity to properly encode cookie information into the "cookie" message header, including any character escaping.

### 4.2 Headers

The `<header/>` element only provides those message headers used for an HTTP request. There is currently no mechanism to convey HTTP response headers.

The name and value attributes from `<header/>` MUST NOT be "pre-escaped" for HTTP message

<sup>4</sup>RFC 2109: HTTP State Management Mechanism <http://tools.ietf.org/html/rfc2109>.

headers. It is the responsibility of the receiving entity to properly encode message headers, including any character escaping.

## 5 Formal Description

### 5.1 Overview

The namespace for this structure is "http://jabber.org/protocol/url-data/scheme/http". Because this namespace is intended to supplement the <url-data/> element, this namespace SHOULD be declared within the <url-data/> element using the "http" namespace prefix (as demonstrated in the examples above). This namespace MUST only be applied when the "target" attribute in <url-data/> uses the scheme "http" or "https".

### 5.2 <auth/> Element

The <auth/> element provides authentication information, including scheme. The contents of this element detail any additional parameters for authentication.

The "scheme" attribute describes the HTTP authentication scheme to use. This attribute MUST be present, and MUST be the name of an HTTP authentication scheme (such as "basic").

### 5.3 <auth-param/> Element

The <auth-param/> element provides authentication parameters. This element has attributes for the parameter name and value.

The "name" attribute specifies the (logical) name of the authentication parameter. This attribute MUST be present, and MUST be applicable to the given scheme, although "realm" is always a valid parameter.

The "value" attribute specifies the value for this authentication parameter. This attribute MUST be present, but MAY be empty if a value is not required.

### 5.4 <cookie/> Element

The <cookie/> element provides client cookie data. This element has attributes for the cookie name and value, as well as the persistence details.

The "name" attribute specifies the name of the cookie. This attribute MUST be present, and MUST NOT be empty. It corresponds to the "name" in the "name=value" syntax.

The "value" attribute specifies the value for the cookie. This attribute MUST be present, but MAY be empty if a value is not required. It corresponds to the "value" in the "name=value" syntax.

The "comment" attribute provides the description of the cookie, as would be provided via the

"Set-Cookie" header. This attribute is OPTIONAL.

The "domain" attribute provides the applicable domain of the cookie, as would be provided via the "Set-Cookie" header. This attribute is OPTIONAL.

The "path" attribute provides the applicable HTTP path of the cookie, as would be provided via the "Set-Cookie" header. This attribute is OPTIONAL.

The "max-age" attribute provides the maximum age for the cookie, as would be provided via the "Set-Cookie" header. This attribute is OPTIONAL, and MUST be a non-negative integer. If not specified, the default value is "0".

The "secure" attribute indicates if this cookie is only offered in secure environments, as would be provided via the "Set-Cookie" header. This attribute is OPTIONAL, and MUST be a boolean value of "true" or "false". The default value is "false".

The "version" attribute provides the version of the cookie, as would be provided via the "Set-Cookie" header. This attribute is OPTIONAL, and SHOULD be "1.0" unless otherwise needed. If not specified, the default value is "1.0".

## 5.5 <header/> Element

The <header/> element provides generic request and entity header information, to be included in the connection attempt. This element is ONLY for message headers included in the HTTP request; it SHOULD NOT be used for message headers included in the HTTP response. The attributes for this element are the name and value of the header.

The "name" attribute specifies the name of the header. This attribute MUST be present, and MUST conform to the "field-name" pattern for RFC 2616. It corresponds to the "name" in the (simplified) "name: value" header syntax.

The "value" attribute specifies the value of the header. This attribute MUST be present, and MUST conform to the "field-value" pattern for RFC 2616. It MAY empty if a value is not required. It corresponds to the "value" in the (simplified) "name: value" header syntax.

## 5.6 XML Schema

```
<?xml version='1.0' encoding='UTF-8'?>
<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  targetNamespace='http://jabber.org/protocol/url-data/scheme/http'
  xmlns='http://jabber.org/protocol/url-data/scheme/http'
  elementFormDefault='qualified'>
  <xs:element name='auth'>
    <xs:complexType>
      <xs:attribute name='scheme' use='required' type='xs:string' />
      <xs:element name='auth-param' minOccurs='0' maxOccurs='unbounded' />
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<xs:complexType>
  <xs:simpleContent>
    <xs:extension base='xs:string'>
      <xs:attribute name='name' use='required' type='xs:string'
        '/>
      <xs:attribute name='value' use='required' type='
        xs:string' />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
</xs:element>
</xs:complexType>
</xs:element>

<xs:element name='cookie'>
  <xs:complexType>
    <xs:attribute name='name' use='required' type='xs:string' />
    <xs:attribute name='value' use='required' type='xs:string' />
    <xs:attribute name='comment' use='optional' type='xs:string' />
    <xs:attribute name='domain' use='optional' type='xs:string' />
    <xs:attribute name='max-age' use='optional' type='xs:integer' />
    <xs:attribute name='path' use='optional' type='xs:string' />
    <xs:attribute name='secure' use='optional' type='xs:boolean' />
    <xs:attribute name='version' use='optional' type='xs:string' />
  </xs:complexType>
</xs:element>

<xs:element name='header'>
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base='xs:string'>
        <xs:attribute name='name' use='required' type='xs:string' />
        <xs:attribute name='value' use='required' type='xs:string' />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

</xs:schema>
```

## 6 Security Considerations

### 6.1 Authentication Information

This document allows complete authentication information to be passed. This information is only as secure as the connection-path between the provider and acceptor.

## 7 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#)<sup>5</sup>.

## 8 XMPP Registrar Considerations

The [XMPP Registrar](#)<sup>6</sup> shall register the "http://jabber.org/protocol/url-data/scheme/http" namespace.

---

<sup>5</sup>The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.

<sup>6</sup>The XMPP Registrar maintains a list of reserved protocol namespaces as well as registries of parameters used in the context of XMPP extension protocols approved by the XMPP Standards Foundation. For further information, see <http://xmpp.org/registrar/>.