

## Table of Contents

<b>Overview .....</b>	<b>2</b>
<b>License .....</b>	<b>2</b>
<b>Features .....</b>	<b>2</b>
<b>Ant task XML elements .....</b>	<b>3</b>
<b>&lt;http&gt; .....</b>	<b>3</b>
supported parameters .....	3
example .....	3
<b>&lt;http&gt; nested elements .....</b>	<b>3</b>
<credentials> .....	3
<keystore> .....	5
<headers>/<header> .....	5
<query>/<parameter> .....	6
<entity> .....	6
<b>Ant configuration .....</b>	<b>7</b>

## Overview

The missing link ant http task was created due to the lack of a full featured, usable and liberally licensed ant HTTP task. This ant task is coded from scratch utilizing only core Java classes; as such it does not require any third party dependencies at runtime.

This ant task also simplifies common aspects of HTTP communication, which other libraries seem to needlessly complicate, such as: authentication, TLS/SSL and HTTP methods other than GET and POST.

## License

The missing link ant http task is licensed under the Apache 2.0 license, a copy of the license can be found with the missing link ant http task distribution or at <http://www.apache.org/licenses/LICENSE-2.0.html>.

## Features

The missing link ant http task was created with the following features in mind:

- No third party library dependencies.
- TLS/SSL support on a configuration, per-connection basis, not JVM-wide.
- Support for HTTP methods GET, POST, PUT, OPTIONS, HEAD and TRACE.
- Support for BASIC authentication.
- Support for multiple URL/URI building options.
- Support for specifying arbitrary HTTP request headers.
- Support for specifying a request entity in-line or through a file.
- Options on what information to print to the output.
- Options to govern what response status codes are expected, and what should cause a build-failing exception.

## Ant task XML elements

### <http>

#### supported parameters

Name	Description	Required	Default	Example
<b>url</b>	HTTP URL	Yes		<a href="http://www.google.com">http://www.google.com</a>
<b>method</b>	HTTP method	No	GET	GET, PUT, POST etc.
<b>printrequest</b>	Print request entity	No	false	true or false
<b>printresponse</b>	Print response entity	No	false	true or false
<b>printrequestheaders</b>	Print req. headers	No	true	true or false
<b>printresponseheaders</b>	Print resp. headers	No	true	true or false
<b>expected</b>	Expected HTTP status	No	200	200, 201, 404 etc.
<b>failonunexpected</b>	Fail on unexpected status	No	true	true or false
<b>outfile</b>	Write response to file	No		Any filename
<b>statusProperty</b>	Property to save status to	No		statusProperty="http.status"
<b>update</b>	Update/overwrite outfile	No	true	true/false
<b>entityProperty</b>	Write response entity to prop	No		response.entity

#### example

```
<http url="http://google.com" method="GET" printrequest="false" printresponse="true"
printrequestheaders="true" printresponseheaders="true" expected="200"
failonunexpected="false" statusProperty="http.status"/>
```

### <http> nested elements

#### <credentials>

#### supported parameters

#### 4 MISSING LINK ANT HTTP TASK

Name	Description	Required	Default	Example
<b>username</b>	BASIC username	Yes		john.doe
<b>password</b>	HTTP password	No		Password
<b>show</b>	Print the credentials	No	false	true/false

#### *example*

```
<http url="http://google.com">  
  <credentials username="john.doe" password="password"/>  
</http>
```

**<keystore>***supported parameters*

Name	Description	Required	Default	Example
<b>file</b>	KeyStore file	Yes		/path/to/keystore.jks
<b>password</b>	KeyStore password	No		password

*example*

```
<http url="http://google.com">
  <keystore file="/path/to/keystore.jks" password="password"/>
</http>
```

**<headers>/<header>***supported parameters*

Name	Description	Required	Default	Example
<b>name</b>	Header name	Yes		Accept
<b>value</b>	Header value	No		application/xml,*/*

*example*

```
<http url="http://google.com">
  <headers>
    <header name="Accept" value="application/xml,*/*" />
    <header name="Content-Type" value="application/xml" />
  </headers>
</http>
```

**<query>/<parameter>***supported parameters*

Name	Description	Required	Default	Example
<b>name</b>	Query parameter name	Yes		qp
<b>value</b>	Query parameter value	No		value

*example*

```

<http url="http://google.com">
  <query>
    <parameter name="qp1" value="value1"/>
    <parameter name="qp2" value="value2"/>
  </query>
</http>

```

**<entity>***supported parameters*

Name	Description	Required	Default	Example
<b>file</b>	File to read entity from	No		request.xml
<b>binary</b>	Treat entity as binary	No	false	true/false
<b>value</b>	Value to use as entity	No		\${my.prop} (for property expansion)

*examples*

```

<http url="http://google.com">
  <entity file="request.zip" binary="true"/>
</query>

<http url="http://google.com">
  <entity><![CDATA[Request Entity]]></entity>
</query>

```

## Ant configuration

The following is a basic example of how to import and use the missing link ant http task:

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="ml-ant-http" basedir="." default="http-get">

  <property name="ml-ant-http.jar" value="ml-ant-http-1.0.jar"/>

  <fileset id="runtime.libs" dir=".">
    <include name="{ml-ant-http.jar}"/>
  </fileset>

  <path id="runtime.classpath">
    <fileset refid="runtime.libs"/>
  </path>

  <taskdef name="http" classname="org.missinglink.ant.task.http.HttpClientTask">
    <classpath refid="runtime.classpath"/>
  </taskdef>

  <target name="http-get">
    <http url="http://www.google.com"/>
  </target>

</project>
```