

# Using Apache with mod\_proxy

中文标题【使用 Apache 和 mod\_proxy】



Atlassian 应用允许用户使用反向代理，但是 Atlassian 并不会为这个功能的配置提供支持。因此，Atlassian **也不能保证能够为这些配置提供任何支持。**

如果你在配置上有任何需求，请参考[Atlassian Answers](#) 中的内容来获取帮助。

这个页面描述了一个使用 Apache HTTP Server 2.4 来反向代理标准的 Tomcat 容器的方法。你也可以在 [NGI NX](#) 中找到类似的配置方法。

你可能在下面的情况中需要使用这个配置：

- 你已经有一个存在的 Apache 网站，同时还希望添加 Confluence 到你已经存在的网站中（例如，<http://www.example.com/confluence>）。
- 你已经有多于一个 Java 的应用了，每一个应用都运行在自己的服务器上，并且使用不同的端口在运行，例如，<http://example:8090/confluence>和<http://example:8080/jira>同时你希望你的这个 2 个应用都能运行在常用的 HTTP（80）端口（例如，<http://www.example.com/confluence>和<http://www.example.com/jira>）。每一个应用都能够被独立的启动，管理和编辑。

**备注：**这个页面的文档是有关 Apache 的配置而不是 Confluence 的配置。Atlassian 将会为你在 Confluence 中的配置提供支持，但是我们不能保证能够帮你解决所有有关 Apache 配置的时候遇到的问题。请意识到，这个页面中的内容只是为配置向你提供相关的的信息，如果参考这个页面配置产生的任何问题，后果自负。

## 基本配置



在这些示例中，我们使用下面的信息：

<http://www.example.com/confluence> - 你计划使用的 URL

<http://example:8090/> - Confluence 当前安装的主机名和端口

<http://example:8091/> - Synchroony 当前安装的主机名和端口，这个服务被用来提供默认的协同编辑服务

</confluence> - 这个是有你 Confluence 计划使用的上下文（在主机名和端口的后面）

</synchroony> - Synchroony 使用的上下文路径，这个提供了协同编辑服务

你需要将上面的 URL 替换成你自己的 URL。

## 1 设置上下文路径

**!**如果你不需要使用上下文来访问 Confluence，例如你希望使用[www.example.com](http://www.example.com)来访问这个，你可以跳过这部分的内容。

设置你 Tomcat 中 Confluence 上下文的路径（在主机名和端口的后面）。在这个示例中，上下文路径被配置为</confluence>

编辑<installation-directory>conf/server.xml"Context" 的定义：

```
<Context path="" docBase="../confluence" debug="0" reloadable="true">
```

然后修改为：

```
<Context path="/confluence" docBase="../confluence" debug="0" reloadable="true">
```

在这个示例中，我们已经使用了</confluence>为上下文的路径。请注意，你不能使用</resources> 为你的上下文路径，这是因为这个上下文的路径在 Confluence 中被用来定义资源，如果你使用这个配置的话，将会在 Confluence 中导致问题。

重启 Confluence，然后你可以尝试使用<http://example:8090/confluence> 进行访问，并确保你能正确访问。

## 2 设置 URL 重定向

下一步，设置 URL 重定向。在<installation-directory>conf/server.xml文件中，使用示例的连接为起始点。

注释掉默认的连接（针对不使用代理的访问）。

本页中的内容：

- [基本配置](#)
  - [1 设置上下文路径](#)
  - [2 设置 URL 重定向](#)
  - [3 配置 mod\\_proxy](#)
  - [4 重启 Apache](#)
  - [5 禁用 HTTP 压缩](#)
  - [6 修改 Confluence 基础 URL](#)
- [添加 SSL](#)
- [更多信息](#)

在 XML 的注释中是以<!-- 开头和--> XML

同时 <!-- and--> 中的内容是默认的连接器，应该看起来如下。

```
<!--
=====
DEFAULT - Direct connector with no proxy, for unproxied HTTP access to Confluence.
=====
-->
<!--
<Connector port="8090" connectionTimeout="20000" redirectPort="8443"
    maxThreads="48" minSpareThreads="10"
    enableLookups="false" acceptCount="10" debug="0" URIEncoding="UTF-8"
    protocol="org.apache.coyote.http11.Http11NioProtocol"/>
-->
```

在HTTP - Proxying Confluence via Apache or Nginx over HTTP头部列出的内容，取消注释连接器。

取消注释的部分，你只需要删除<!--和-->包含的连接器注释就可以了。

下面显示的是默认连接器被注释掉，但是 HTTPS 连接器没有被注释。头部同时保留注释。

```
<!--
=====
DEFAULT - Direct connector with no proxy, for unproxied HTTP access to Confluence.
=====
-->
<!--
<Connector port="8090" connectionTimeout="20000" redirectPort="8443"
    maxThreads="48" minSpareThreads="10"
    enableLookups="false" acceptCount="10" debug="0" URIEncoding="UTF-8"
    protocol="org.apache.coyote.http11.Http11NioProtocol"/>
-->
<!--
=====
HTTP - Proxying Confluence via Apache or Nginx over HTTP
=====
-->
<Connector port="8090" connectionTimeout="20000" redirectPort="8443"
    maxThreads="48" minSpareThreads="10"
    enableLookups="false" acceptCount="10" debug="0" URIEncoding="UTF-8"
    protocol="org.apache.coyote.http11.Http11NioProtocol"
    scheme="http" proxyName="<subdomain>.<domain>.com" proxyPort="80"/>
```

在上面显示内容的最下面，插入你的proxyName和proxyPort：

```
<Connector port="8090" connectionTimeout="20000" redirectPort="8443"
    maxThreads="48" minSpareThreads="10"
    enableLookups="false" acceptCount="10" debug="0" URIEncoding="UTF-8"
    protocol="org.apache.coyote.http11.Http11NioProtocol"
    scheme="http" proxyName="www.example.com" proxyPort="80"/>
```



如果你计划启用 HTTPS，使用HTTPS - Proxying Confluence via Apache or Nginx over HTTPS。下面的连接器。

### 3 配置 mod\_proxy

使用下面的一个示例来编辑你的http.conf文件中有关应用服务器代理的配置。

⚠ 你需要针对你的 Apache 启用下面的一些模块，如果这些模块还没有在你的 Apache 中启用的话：

- mod\_proxy
- mod\_proxy\_http

- proxy\_wstunnel
- mod\_rewrite

(proxy\_wstunnel 和 mod\_rewrite 是针对 Confluence 6.0 需要的新的模块)

http.conf文件的格式和模块的位置路径，针对你使用操作系统的不同可能有所不同。你们推荐 Windows 的用户在这里配置的时候使用绝对路径。

### 示例 1：配置上下文路径

在这个示例中，如果你已经在上面的第一步中设置了上下文路径，同时你的 Confluence 服务器也可以通过配置的上下文路径进行访问，例如这个<http://www.example.com/confluence>。

在这个示例中，用户将会连接到 Synchrony，这个是针对协同编辑所使用的服务，是通过 WebSockets 直连的。

**⚠**在配置文件中的配置顺序是非常重要的。

#### Apache HTTP server 2.4

```
# Put this after the other LoadModule directives
LoadModule proxy_module /usr/lib/apache2/modules/mod_proxy.so
LoadModule proxy_http_module /usr/lib/apache2/modules/mod_proxy_http.so
LoadModule proxy_wstunnel_module /usr/lib/apache2/modules/mod_proxy_wstunnel.so
LoadModule rewrite_module /usr/lib/apache2/modules/mod_rewrite.so

# Put this in the main section of your configuration (or virtual host, if using Apache virtual hosts)
ProxyRequests Off
ProxyPreserveHost On

<Proxy *>
    Require all granted
</Proxy>

ProxyPass /synchrony http://<domain>:8091/synchrony
<Location /synchrony>
    Require all granted
    RewriteEngine on
    RewriteCond %{HTTP:UPGRADE} ^WebSocket$ [NC]
    RewriteCond %{HTTP:CONNECTION} Upgrade$ [NC]
    RewriteRule .* ws://<domain>:8091%{REQUEST_URI} [P]
</Location>

ProxyPass /confluence http://<domain>:8090/confluence
ProxyPassReverse /confluence http://<domain>:8090/confluence

<Location /confluence>
    Require all granted
</Location>
```

**注意：**如果你使用的是 HTTP Server 2.2，但是你的 Confluence 使用的 6.0 及其后续版本，这个配置是不能成功的。如果你计划使用 SSL，你需要使用 2.4.10 或者后续版本。

### 示例 2：不使用上下文的配置

在这个示例中，如果你已经跳过了第一步的配置，没有使用上下文路径进行访问，例如访问的 URL 为<http://www.example.com/>。

在这个示例中，用户将会连接到 Synchrony，这个是针对协同编辑所使用的服务，是通过 WebSockets 直连的。

**⚠**在配置文件中的配置顺序是非常重要的。

**Apache HTTP server 2.4**

```
# Put this after the other LoadModule directives
LoadModule proxy_module /usr/lib/apache2/modules/mod_proxy.so
LoadModule proxy_http_module /usr/lib/apache2/modules/mod_proxy_http.so
LoadModule proxy_wstunnel_module /usr/lib/apache2/modules/mod_proxy_wstunnel.so
LoadModule rewrite_module /usr/lib/apache2/modules/mod_rewrite.so

# Put this in the main section of your configuration (or virtual host, if using Apache virtual hosts)

ProxyRequests Off
ProxyPreserveHost On

RewriteEngine On
RewriteCond %{REQUEST_URI} !^/synchrony
RewriteRule ^/(.*) http://<domain>:8090/$1 [P]

<Proxy *>
    Require all granted
</Proxy>

ProxyPass /synchrony http://<domain>:8091/synchrony

<Location /synchrony>
    Require all granted
    RewriteEngine on
    RewriteCond %{HTTP:UPGRADE} ^WebSocket$ [NC]
    RewriteCond %{HTTP:CONNECTION} Upgrade$ [NC]
    RewriteRule .* ws://<domain>:8091%{REQUEST_URI} [P]
</Location>

ProxyPass / http://<domain>:8090
ProxyPassReverse / http://<domain>:8090

<Location />
    Require all granted
</Location>
```

**注意：**如果你使用的是 HTTP Server 2.2，但是你的 Confluence 使用的 6.0 及其后续版本，这个配置是不能成功的。如果你计划使用 SSL，你需要使用 2.4.10 或者后续版本。

## 4 重启 Apache

为了让新的配置生效，你需要运行下面的命令来让 Apache 重启后载入新的配置文件：

```
sudo apachectl graceful
```

## 5 禁用 HTTP 压缩

如果在代理和 Tomcat 中启用了压缩，这个将会在整合其他 Atlassian 应用的时候出现问题，例如 JIRA。请禁用 HTTP 压缩，并请参考 [Compressing an HTTP Response within Confluence](#) 页面中的内容。

## 6 修改 Confluence 基础 URL

最后一个步骤是配置 [Base URL](#) 的地址来指向你没有使用的代理的地址，例如 <http://www.example.com/confluence>。

## 添加 SSL

如果你计划在你的应用中启用 SSL，请参考 [Securing your Atlassian applications with Apache using SSL](#) 页面中的内容，并确定你在相同的连接器中选择 HTTPS。

## 更多信息

- [mod\\_proxy\\_html](#) 站点提供了一些示例来为你如何配置反向代理提供帮助。
- [Apache Week has a tutorial](#) 这个指南提供了如何解决复杂的配置情况和 ProxyHTMLURLMap。