

Wicket application behind mod_proxy_http and https

Wicket usually redirects a lot. It, by default, uses the [redirect after post pattern](#) to solve the double submit problem.

Suppose the Wicket application is running in a servlet container (like Tomcat or Jetty).

That servlet container is fed requests through a [Apache HTTPD](#) the [mod_proxy_http module](#).

Apache HTTPD is configured to serve clients using HTTPS, which is terminated at the HTTPD server before it is forwarded tot the servlet container.

The servlet container knows nothing about the fact that HTTPS is used so each redirect that is generated will look like [http://servername.com/someuri](#) instead of [https://servername.com/someuri](#).

Therefore the Header directive in a configuration file for Apache HTTPD is typical in Apache HTTPD for a wicket application running in a servlet container behind Apache:

```
ProxyPreserveHost On
ProxyPass /contextroot http://localhost:9095/contextroot
ProxyPassreverse /contextroot http://localhost:9095/contextroot
Header edit Location ^http://someservername.com/contextroot/ https://someservername.com/contextroot/
```

It looks like this is the logical place for manipulation of the URL since this is also the place where SSL is terminated (HTTPS stops at Apache HTTPD and HTTP is then forwarded to the servlet container).

This is useful if all of your application must be served over HTTPS.

Wicket has [some facilities for HTTPS](#) but that is useful only if some but not all of your pages must be served over HTTPS.