

Running Unimus behind a HTTP(S) proxy

Unimus requires outbound HTTPS connectivity to the licensing server.

Some environments might not allow outbound connections directly, and require usage of HTTP(S) proxies.

Running Unimus behind a HTTP(S) proxy is fully supported, but requires additional configuration.

On Windows:

For portable:

Create a file named **Unimus.I4j.ini** in the same directory as the Unimus executable.

Change **Unimus** in the above file name to exactly match the name of the Unimus executable.

For installer:

Add the below lines to **Unimus.I4j.ini** in 'C:\Program Files\Unimus\'.

Inside of this configuration file, please add the following lines:

```
-Dhttp.proxyHost=1.1.1.1  
-Dhttp.proxyPort=8080  
-Dhttps.proxyHost=2.2.2.2  
-Dhttps.proxyPort=8443
```

Please make sure to place each argument into its own line.

Insert proper values behind each configuration argument.

If you only use HTTP proxy, or only use HTTPS proxy, remove the other from the configuration file.

Restart Unimus, and any HTTP(S) outbound connectivity from Unimus will go through the configured proxy.

On Linux (Debian/Ubuntu):

Edit the contents of **/etc/default/unimus**.

Add this into the file:

```
-Dhttp.proxyHost=1.1.1.1 -Dhttp.proxyPort=8080 -Dhttps.proxyHost=2.2.2.2  
-Dhttps.proxyPort=8443
```

Insert proper values behind each configuration argument.

If you only use HTTP proxy, or only use HTTPS proxy, remove the other from the configuration file.

Restart Unimus, and any HTTP(S) outbound connectivity from Unimus will go through the configured proxy.

HTTPS proxy type

When using a HTTPS proxy, Unimus supports 2 proxying modes:

- HTTP 1.1 Connect
- SOCKS

By default, the "HTTP 1.1 Connect" proxy method will be used. If your proxy requires SOCKS proxying, you can configure this by:

```
-Dhttps.proxyType=socks
```

Proxies with authentication

If you need to authenticate to your proxy, you will need to setup additional configuration:

```
-Dhttp.proxyUser=username  
-Dhttp.proxyPassword=password  
-Dhttps.proxyUser=username  
-Dhttps.proxyPassword=password
```

If your proxy uses Basic Auth, this is disabled by default. You will also need to configure:

```
-Djdk.http.auth.tunneling.disabledSchemes=" "
```

(please keep proper format for Linux / Windows)

Running in Tomcat:

Please refer to Tomcat documentation on configuring proxying.

How this is configured depends on the distribution and packaging of your Tomcat.