

Installing WebSpellChecker on Linux with Apache Tomcat

This step-by-step guide outlines the main steps for the WebSpellChecker Server installation on a Linux-based server with Apache Tomcat Application Server. In this guide we come through the whole installation process from downloading installation packages to embedding spell/grammar checking functionality to your web-based system.



Before you begin:

1. Check if your environment configuration meets:
 - a. [WebSpellChecker Server System Requirements](#)
 - b. [WebSpellChecker Server Hardware Requirements](#)
2. Acknowledge with [WebSpellChecker Server Architecture Diagram](#).
3. Do you have a valid license?

First of all you need to get a valid License Ticket ID to proceed with the license activation. Depending on the server license type, it may be a commercial (e.g. 1 year) or a 30-day trial license. Except the validity period, there is no difference between ticket types from the technical perspective.

Good to know:

- If you do not have a license yet, you can obtain it using one of the following ways:
 - request a 30-day trial license for evaluation and testing purposes;
 - acquire a commercial license by contacting our sales department.
- Starting from v5.8.1 released on December 30, 2020, the license is no longer validated by or tied to the hardware characteristics of a machine or server. License for older versions of the packages is hardware-dependent.
- If you are migrating WebSpellChecker from one server to another, you will need to deactivate (detach) a license on the current server and then reactivate it on a new one. Check carefully the **Migrating License to New Server** guide under the Licensing section.
- If you have problems with your license, contact us.

1. [Download Installation Package](#)
2. [Install WebSpellChecker](#)
Option A: Binding of Static Files and Serving Service Requests by AppServer
Option B: Serving both Static Files and Service Requests by Java Application Server (deployment of wscservice.war)
3. [Activate License](#)
4. [Finalize Installation](#)
5. [Configure Grammar Engine](#)
6. [Verify Operability](#)
[WebSpellChecker Server Status](#)
[WebSpellChecker Server Version](#)
[WebSpellChecker Demo Samples](#)
7. [Integrate WebSpellChecker](#)
Option A. [Integration Guides](#)
Option B. [Demo Samples](#)

1. Download Installation Package

Download WebSpellChecker Server installation package for the corresponding version of your operating system (OS). By default, it is provided on a license purchase or renewal. You can also request it by [contacting us](#).

2. Install WebSpellChecker

2.1. Unpack the WebSpellChecker Server installation package (e.g. wsc_app_x64_5.5.x.x_xx.tar.gz).

```
$ tar -xvf wsc_app_x64_5.5.x.x_xx.tar.gz
```

2.2. Switch to the directory with the unpacked installation package and execute **automated_install.pl** script.

```
$ sudo perl automated_install.pl
```



If you are running the installation without root privileges, make sure all necessary directories have **write** permissions for the correct installation process.

2.3. Before proceeding with installation, read and confirm the terms and conditions of [WebSpellChecker Software License Agreement](#).

Welcome to the WebSpellChecker (WSC) application installation wizard!

Before you go ahead with the installation, make sure that your environment configuration meets the WSC application system requirements.

If you are all set, please follow carefully the wizard instructions.

PLEASE MAKE SURE YOU ARE ACKNOWLEDGED AND ACCEPT THE WEBSPELLCHECKER SOFTWARE LICENSE AGREEMENT TERMS AND CONDITIONS PUBLISHED ON THE WEBSPELLCHECKER OFFICIAL WEBSITE. BY DOWNLOADING OR INSTALLING THIS SOFTWARE, YOU ACCEPT THE FULL TERMS OF THIS AGREEMENT.

IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT, YOU MUST NOT INSTALL OR USE THE WEBSPELLCHECKER SOFTWARE.

2.4. Select installation path. The default installation path is **/opt/WSC**. However, you may decide to change it and set a desired one. As soon as you defined the installation path, press **Enter** to proceed.

Step 1: Check Configuration Settings.

Install path [/opt/WSC]:

2.5. Select the products you want to be installed. By default, **WProofreader** (Option 1) is selected.

Step 2: Select Products.

1. WProofreader(HTML Editable Controls, CKEditor 4/5, Froala Editor, TinyMCE, Quill, etc.)
2. Web API
3. SCAYT Plugin for CKEditor 4
4. WSC Dialog Plugin for CKEditor 4
5. Select all products

List the products you want to install. Use the comma as separator. [1]:

2.6. Confirm whether you want Demo Samples and Documentation to be installed. If so, press **Enter** to proceed.

Step 3: Install Demo Samples and Documentation

Do you want to install Demo Samples and Documentation? Enter 'y' or 'n' [y]:

2.7. During this step the installer copies files to the selected installation directory.

Step 4: Copy the WSC Files to the Installation Directory.

2.8. Define permissions. Set appropriate permissions for the application. Define the files' ownership, a user and group in the **user:group** format (e.g. 33:33).

Step 5: Define the WSC Application Permissions.

Specify the files ownership, a user and group in the 'user:group' format.

It can be the same user and group which are used for the Apache HTTP Server, i.e. 'www-data:www-data' or '33:33'.

Specify the files owner as user:group [nobody:nobody]: 33:33

* Setting the WSC application permissions.
You specified 33:33

2.9. Select the type of the protocol (HTTP or HTTPS) that will be used for communication with the WebSpellChecker service and a web server or Java application server. By default, HTTPS (Option 1) is selected. Press **Enter** to confirm the selection.



You will have this step if you selected products that can communicate with AppServer directly (for example, WProofreader or Web API). The exception is WSC Dialog Plugin for CKEditor 4. In this case all the requests and static files will be served by a selected web server or application server via the FastCGI protocol (SSRV.CGI component).

Step 6: Select Protocol Type.

Select the type of the protocol that will be used for communication with the WebSpellChecker service and a web server or application server.

If you plan to use HTTPS, please make sure you have a valid SSL certificate.

1. HTTPS
2. HTTP

Input the number from 1 to 2 [1]:

2.10. If you have selected HTTPS on the previous step, specify the path to a valid SSL certificate file (*.pem) and its private key file (*.pem) to establish a secure connection. Otherwise, this step is not available.

Step 7: Configure Service SSL Certificate.

Specify the path to the SSL certificate file (*.pem): /cert.pem

Specify the path to the private key file (*.pem): /key.pem

If your SSL certificate is password-protected, provide the password.

If your SSL certificate is password-protected, please specify the password. Otherwise, leave it empty:

2.11. Specify the WebSpellChecker AppServer service port. The default port of AppServer is 2880. Press **Enter** to proceed the installation using port 2880.



Don't use the port number of your web server or Java application server here. The configuration settings for it are available further.

Specify the service port (by default: 2880).

If you prefer to use a custom port, make sure it is not in use by another service. [2880]:

- Choose **Java Application Server (Apache Tomcat, Oracle WebLogic Server, Oracle GlassFish Server, etc.)** option to serve static files from WebSpellChecker Server package on Apache Tomcat Application Server (Option 2) and press **Enter**.

Step 8: Choose Web or Application Server type.

Select a particular web or application server which will be used for the WSC application server running.

1. Apache HTTP Server
2. Java Application Server (Apache Tomcat, Oracle WebLogic Server, Oracle GlassFish Server, etc.)
3. Other

Input the number from 1 to 3: 2



This section outlines the details of **wscservice.war** deployment on Apache Tomcat (static files and all requests to engines are served) or binding only static files and the rest of the service requests such as spelling or grammar checking, directly connecting with AppServer.

Option A: Binding of Static Files and Serving Service Requests by AppServer

To bind only static files and the rest of the service requests such as spelling or grammar checking directly connecting with AppServer:

1. Stop Apache Tomcat before modifying the configuration files.
2. Add the following paths to the **server.xml** file inside the **<Host name="localhost" appBase="webapps" unpackWARs="true" autoDeploy="true">** tag:

```
<Context docBase="/opt/WSC/WebComponents/WebInterface" path="/wscservice" />
<Context docBase="/opt/WSC/WebComponents/Samples" path="/wscservice/samples" />
```

3. If you plan to host WebSpellChecker and your web app(s) on different domains, you might face the CORS issues. To avoid this, add the **Access-Control-Allow-Headers** to enable cross-origin requests to the Apache Tomcat settings. For this you can use [CORS filter](#).
4. Start Apache Tomcat after modifying the configuration files.

Option B: Serving both Static Files and Service Requests by Java Application Server (deployment of wscservice.war)

To deploy wscservice.war on Apache Tomcat for binding static files and all service requests to engines such as spelling or grammar checking :

1. Set and check the **WSC_DIR** environment variable. It must be pointed to the WebSpellChecker Server installation directory (for example, **WSC_DIR=/opt/WSC** as in example below):

```
export WSC_DIR=/opt/WSC
echo @WSC_DIR
```

2. Deploy **wscservice.war** on your Apache Tomcat Application Server.

- Copy the **wscservice.war** file from <WSC_DIR>/J2EEServerComponents to the Apache Tomcat **/webapps** directory.
- Start Apache Tomcat, or stop and then start if it is already running.



Note that the **wscservice.war** file should be unpacked automatically. Otherwise, unpack it manually.

2.12. Define settings of the selected web server or application server.

- Specify a domain name. The default value is **localhost**. Either enter your domain name or confirm the default value by pressing Enter.

Step 9: Specify Web Server or Application Server Settings.

Specify a domain name, the service virtual directory and a port number that will be used by a web server or application server.

Domain name [localhost]:

- Specify a port number of a web server or Java application server. The installer will suggest the port number based on the specified protocol and selected web server or Java application server.

Web or application server port [8443]:

- Provide the name of the virtual directory for the WebSpellChecker Service. The default value is **wscservice**.

Service virtual directory [wscservice]:

On this step the installer summarizes the information that you have entered on the previous steps.

```
* PROTOCOL = https
* DOMAIN NAME = localhost
* SERVICE PORT = 2880
* WEB / APPLICATION SERVER PORT = 8443
* VIRTUAL DIRECTORY = wscservice
* SERVICE URL = https://localhost:8443/wscservice
```

3. Activate License

On this step you need to choose whether to run the license activation process. If you want to go ahead with the license activation, press **Enter**, then specify the **License Ticket ID**, and press **Enter** to activate the license automatically.

Step 10: Activate License.

Do you want to proceed with the license activation? Enter 'y' or 'n' [y]:
Specify your License Ticket ID: Your_License_Ticket_ID

Do you want the installer to attempt to activate your license automatically?
The automatic activation requires the Internet connection available on this machine.
Otherwise, please proceed with the manual activation instead. Enter 'y' or 'n'. [y]:

If you already have a license activated or you prefer to proceed with the license activation step later, you can skip this step.

Visit the [License activation on Linux](#) guide for more details on how to activate a license on a Linux server.

4. Finalize Installation

If no errors happened during the setup, and everything is configured properly, you will receive a confirmation that the application has been installed.

Congratulations! The WSC application has been successfully installed to /opt/WSC directory.

5. Configure Grammar Engine

If the path to Java Virtual Machine (JVM) which is required for the Grammar engine is not auto-detected, the script will ask you to specify the correct path to JVM again.

Step 11: Specify Path to Java Virtual Machine.

At least Java 8 is required to use the Grammar engine. Please specify the path to Java Virtual Machine (JVM). You can skip this step and add the path to JVM manually in the '/opt/WSC/AppServer/AppServerX.xml' file (the PathToJavaVirtualMachine tag).

Do you want to specify a path to JVM? Enter 'y' or 'n' [y]:

You have an option to **skip** this step during the setup process and set a path to JVM manually. Refer to [Enabling Grammar Engine](#) guide to find out more.

6. Verify Operability

Now you are ready to check if the installation process of the application went well, and all the components are functioning properly.

Step 12: Verify the WSC Application Operability.

Check the application version, status, and access to the demo samples to verify if the application works properly after the installation.

a. Version:

<https://localhost:2880/?cmd=ver>

b. Status:

<https://localhost:2880/?cmd=status>

c. Default demo samples:

<https://localhost/wscservice/samples/>

Find the correct responses for the application version and status below.

WebSpellChecker Server Status

Here is a correct status response of WebSpellChecker Server:

```
{
  "SpellCheckEngine": {
    "active": true
  },
  "GrammarCheckEngine": {
    "active": true,
  },
  "EnglishAIModel": {
    "active": true
  },
  "GermanAIModel": {
    "active": true
  },
  "SpanishAIModel": {
    "active": true,
  },
  "EnglishAutocomplete": {
    "active": true,
  }
}
```

WebSpellChecker Server Version

Here is a correct version response for WebSpellChecker Server 5.x.x.x on a Linux server:

```
{
  "Copyright": "(c) 2000-2023 WebSpellChecker LLC",
  "ProductWebsite": "webspellchecker.com",
  "ProgramVersion": "5.x.x.x x64 master:xxxxxxx (xxxx) #xx",
  "PackageVersion": "5.x.x.x master:xxxxxxx (xxx) #xx"
}
```

WebSpellChecker Demo Samples

Here is the correct view of the WebSpellChecker demo samples:

**OVERVIEW****HTML EDITABLE CONTROLS**[Multiple Controls](#)**WYSIWYG EDITORS**[Multiple Editors](#)**WProofreader in HTML Editable Controls**[DOCUMENTATION](#)

This sample demonstrates the work of the WProofreader in HTML Editable Controls.

WProofreader can be easily enabled and used with HTML editable controls such as **input** or **textarea** and elements with contenteditable attribute set to 'true'.

WProofreader in Textarea

This **sampl** text demonstrates the work of a new **WProofreader**. Just start typing here and you will see your spelling **and and** grammar mistakes.

3

WProofreader in Iframe

This **sampl** text demonstrates the work of a new **WProofreader**. Just start typing here and you will see your spelling **and and** grammar mistakes.

4

WProofreader in Div

This sampl text demonstrates the work of a new WProofreader. Just start typing here and you will see your spelling and and grammar mistakes.

7. Integrate WebSpellChecker

Integrate WebSpellChecker products into your web application.



Do not use 2880 port when accessing **samples** or other statics files like **wscbundle.js** file as such files will be processed by a selected web server or Java application server. 2880 is an internal port that WSC AppServer listens to and which is designed to process service requests (check spelling, grammar and others).

Option A. Integration Guides

Visit [Products and integrations guides](#) to learn more how to get started with WebSpellChecker products, one of them being [WProofreader](#).

WProofreader

[WProofreader](#) is multilingual proofreading tool for web applications. It enables grammar and spell check while you type or work with your text in a dedicated dialog. It detects editable fields in focus, be it HTML editable controls or WYSIWYG editors, and enables spelling and grammar checking there. View the [full list of supported integrations](#).

WProofreader benefits:

- Available as a standalone secure solution on your server or as an annual subscription to the service hosted by WebSpellChecker.
- Out of the box offer provides spell check for 17 languages and grammar check for 15 out of 17 [default languages](#);
- All modern [web browsers are supported](#) including the latest stable versions of Chrome, Firefox, Safari, MS Edge, and Internet Explorer 11 and mobile web browsers such as Chrome for Android and iOS Safari;
- Over 150 additional languages as well as specialized medical and legal dictionaries are [available to extend the default languages set](#);
- [Global custom dictionaries](#) managed by system admins and intended for all WProofreader users;
- [Personal user dictionaries](#) for end-users;
- [WAI-compliant](#), up to [WCAG 2.1](#) and [Section 508 accessibility standards](#);
- Operation and installation via HTTPS protocol is supported to fulfill strict security requirements.

With the Cloud-based version of WProofreader you can access and use its features without server-side installation. In the Server-based version, all the application data is securely processed on your own server. Refer to the [Get started with WProofreader Server \(autoSearch\)](#) guide for more details.

WProofreader allows users to see and correct spelling and grammar problems on the fly while they are typing directly in the text input field. Every misspelled word found will be underlined with a red wavy line and every grammar problem detected will be marked with a green one respectively. The user just needs to hover a marked word or phrase and replace it with a proper one from the list of given corrections.

Option B. Demo Samples

Depending on the type of elements, namely HTML native elements or rich text editors where the spelling and grammar check is needed, you may choose and use the demo samples that are shipped with the package.

To access the default samples, go to [http\(s\)://your_host_name:port/wscservice/samples/](http(s)://your_host_name:port/wscservice/samples/) where port is the **port** of your web server or Java application server. The default samples contains all the necessary samples and instructions that will help with the integration of the WebSpellChecker products. Moreover, each sample contains detailed development documentation with a list of methods and parameters available.