

# **LICENSE4J AUTO LICENSE GENERATION AND ACTIVATION SERVER USER GUIDE**

**VERSION 1.4.0**



**LICENSE4J**  
[www.license4j.com](http://www.license4j.com)

## Table of Contents

Getting Started.....	2
Server Roles .....	4
Installation .....	6
Server WAR Deployment .....	6
Server with Embedded Tomcat Installation .....	7
Configuration.....	8
Error and Access Logging.....	13
Server License .....	14

## Getting Started

License4J Auto License Generation and Activation Server handles auto license generation requests, activation/deactivation requests, license modification requests, license availability check requests and online license key validation requests.

License4J License Manager GUI tool stores products, templates, and licenses in the defined database server. When a database server other than local embedded derby database is selected in License Manager GUI options, license activation features and product auto license generation settings are enabled.

When an auto license generation request is sent to Auto License Generation and Activation Server, it searches for requested product and template to generate the license. After all security checks performed, it generates the license then sends back as specified in auto license generation output settings for the product.

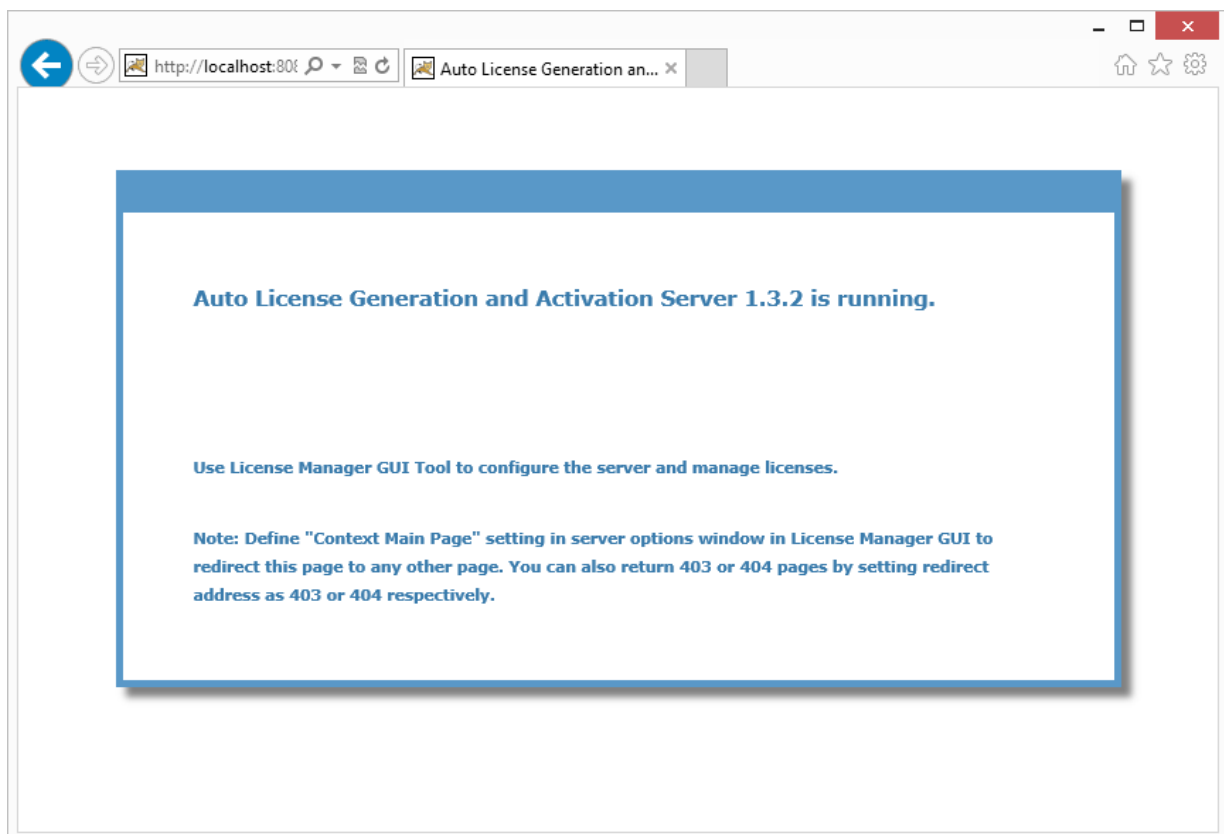
License4J Runtime Library has *autoActivate* method which sends activation request to defined Auto License Generation and Activation Server or Online.License4J system. When an activation request is obtained, requested license is searched in database and if found, allowed activation limit is checked and if activation is allowed, server sends back the activated license text or activation code. If same computer tries to activate license second time, due to e.g. re-installation of software, activation server returns the same license previously activated if hardware ID matches. Therefore when hardware ID does not change re-activation attempts always return the first activation.

License4J defines a new license type which is "Online Basic Key Floating Over Internet". It is very similar to a floating license, but Floating License Server is not required and it works over the Internet. Therefore Auto License Generation and

Activation Server has a role named as "Online Key Validation" for managing online basic keys floating over Internet.

Since License4J Auto License Generation and Activation Server is a web application, it can run on load-balance or fail-over cluster environments.

Auto License Generation and Activation Server does have a management GUI accessible with a browser. The only page that server displays are given in the screenshot below, and License Manager GUI application is the only tool to manage licenses.



## Server Roles

License4J Auto License Generation and Activation Server has following eight roles, you can enable and disable roles as required. With these roles, more than one server can be deployed and run to balance load on servers or meet security needs.

- **License Generation**

If enabled, server accepts auto license generation requests, and generates licenses. URL will start with http or https depending on your server configuration and becomes <http://YourServerName/algas/generate/xxx>, the unique number shown as xxx after generate keyword is displayed in auto license generation settings window in License Manager GUI.

- **Auto License Activation**

If enabled, server will handle auto license activation and modification requests. The URL for auto activations will be <http://YourServerName/algas/autoactivate> and it accepts connections only from Runtime Library *autoActivate* and *autoActivateWithCustomHardwareID* method.

- **Manual License Activation**

If enabled, manual license activation will be allowed, but you should make a form for your customers to submit activation request. Form should be submitted to <http://YourServerName/algas/manualactivate> and should include a hidden value with name *action* and value *activate*; and license activation request string in an input area with name *linfo*. A sample form is given below.

```
<form action="http://YourServerName/algas/manualactivate" method="post">
  <textarea name="linfo"></textarea>
  <input type="hidden" name="action" value="activate">
  <input type="submit" value="submit">
```

```
</form>
```

- **Auto License Deactivation**

If enabled, server will handle auto license deactivation requests. The URL for auto deactivations will be <http://YourServerName/algas/autodeactivate> and it accepts connections only from Runtime Library *autodeActivate* method.

- **Manual License Deactivation**

If enabled, manual license deactivation will be allowed, but you should make a form for your customers to submit deactivation request. Form should be submitted to <http://YourServerName/algas/manualdeactivate> and should include a hidden value with name *action* and value *mdeactivate*; and license deactivation request string in an input area with name *linfo*. A sample form is given below.

```
<form action="http://YourServerName/algas/manualdeactivate"
method="post">
<textarea name="linfo"></textarea>
<input type="hidden" name="action" value="mdeactivate">
<input type="submit" value="submit">
</form>
```

- **Online Key Validation**

If enabled, server will lease online basic keys floating over Internet. The URL for license barrow will be <http://YourServerName/algas/validateobk> and it accepts connections only from Runtime Library *validate* method.

- **License Availability Check**

If enabled, server will respond license availability check requests. Generated licenses or activations can be deleted on server for some reason, and if it is needed to block license owner customer to use license, it can be checked with a single method in Runtime Library, and if license availability checks fails, software product can be disabled. This role can also be used to blacklist some licenses and detect on customers' computers. It can be used to check for

activations. If you enabled license deactivation feature, you can be sure that a deactivated license is not used anymore. The URL for license check will be <http://YourServerName/algas/checkavailability> and it accepts connections only from Runtime Library *validate* method.

- **License Modification**

If enabled, server will allow auto license modification requests. The URL for license modification will be <http://YourServerName/algas/autoactivate> and it accepts connections only from Runtime Library *validate* method. License modification is available for only activated licenses and if activation return method is license text. Activated license text is modified if given modification key is valid, and new modified activated license text is returned to user.

In addition to these eight server roles License4J Auto License Generation and Activation Server also runs a time servlet to provide web time to License4J Runtime Library to check date/time if a license has enabled online date/time check feature. The time source URL is <http://YourServerName/algas/time> and can be defined in online date/time check feature web server field in license generation wizard of License Manager GUI.

## Installation

### Server WAR Deployment

Auto License Generation and Activation Server is tested on Apache Tomcat version 7 with JDK 1.7. All required libraries are included in application WAR file.

Download package includes the application deployment file (WAR file) and a sample configuration file. Copying these two files is sufficient to complete installation.

1. Copy *license4j-server.conf* file to Tomcat configuration directory. Directory name is *conf* and can be found in Tomcat installation path.

2. Open *license4j-server.conf* file and define the only required keyword **jdbc-url** for your database which you connect with License Manager GUI and generate licenses.
3. Copy *algas.war* file to Tomcat *webapps* folder for auto deployment.
4. Copy your database server JDBC driver jar file to Tomcat lib directory.
5. If Tomcat is already started, it will be deployed. If Tomcat is not running, start Tomcat.
6. When application is deployed, it connects to Online.License4J and activates included trial license. After first startup it can be changed on Auto License Generation and Activation server settings window of License Manager GUI.

A log file is created in Tomcat logs directory with name *license4-server.log*, check the log file for errors and information messages. An access log file (*license4-access.log*) is created in the same directory and includes all failed and successful requests.

## Server with Embedded Tomcat Installation

Auto License Generation and Activation Server with embedded Tomcat installation includes both an embedded Tomcat and Java 1.7. Setup file is distributed for Windows and Linux operating systems and installs all required files on operating system; then a sample configuration file is created on first run.

After installation completed, installed service is started and default configuration file is created, required **jdbc-url** keyword must be defined in configuration file. By default server listens on port 8090, you can browse to <http://localhost:8090> and see the server status. If server is not running due to an error in configuration file or database, the main reason for error is displayed on the page.



Configuration file and logs files are located in local *ProgramData* folder (C:\ProgramData\ALGAS) on Windows, and in installation path on Linux.

Created default configuration file has required descriptions for configuration keywords. Different from WAR file deployment, you can define interface and port number on which server will listen, and SSL certificate settings to accept secure connections.

Log file is created in program data folder (C:\ProgramData\ALGAS) on Windows and in installation path on Linux. An access log file (*license4-access.log*) is created in the same directory and includes all failed and successful requests.

## Configuration

License4J Auto License Generation and Activation Server needs only *jdbc-url* keyword to be specified in a configuration file. The configuration file name is *license4j-server.conf* and should be placed in Tomcat configuration directory named as *conf*, and in *ProgramData* (C:\ProgramData\ALGAS) folder on Windows and installation folder on Linux. A sample configuration file is included in download package which includes sample jdbc url definitions for MySQL, Postgresql and MS SQL database servers.

License4J License Manager GUI is used to configure Auto License Generation and Activation Server. *Server settings* menu item is found in *Tools* menu. As in the following screenshot, all server settings are displayed and can be modified.

License4J Auto License Generation and Activation Server is scheduled to check for configuration changes each 10 seconds so after changing any setting, it is not required to restart or redeploy application.

**Auto License Generation and Activation Server Options**

**Auto License Generation and Activation Servers**

Server List:

**Server Information**

Name:

Version:  Hostname:

Description:

Hardware ID:

Tomcat Path:

Context Main Page:

License Key:

Maintenance End:

**Server Roles**

☒ License Generation ☒ Online Key Validation ☒ License Availability Check

☒ Auto License Activation ☒ Manual License Activation ☒ License Modification

☒ Auto License Deactivation ☒ Manual License Deactivation

Name and Description fields are optional and used just for displaying more meaningful information. Server's Hardware ID and license key is displayed and you can change license key with the related button. Context main page setting defines the address which <http://YourServerName/algas/> main application context page will be redirected. If main context page will be redirected to another page address should be given as <http://www.example.com>; if it is required to return a 403 or 404 page, 403 or 404 should be set relatively. e.g. When 403 is given as context main

page, <http://YourServerName/algas/> page will return a 403 forbidden page. *Remove Server* button removes only server settings from database, if removed server is still running it is stopped. When tomcat is restarted on which a removed Auto License Generation and Activation Server deployed, server settings is reset to default trial settings and created on the database.

Auto License Generation and Activation Server sends notification e-mails for selected cases. It can send notifications for every successful and failed request server gets.

*Change Key* button displays a dialog to enter your license key for Auto License Generation and Activation Server. *Enter Modification Key* button is used to enter a modification key to modify current license. It is used to extend maintenance period when a maintenance is purchased.

**Server Notification Settings**

**SMTP Server Information**

SMTP Host:  Port:

From Address:

To Address:

☐ Use TLS

☐ Require Authentication

Username:

Password:

**License Notifications**

<input type="checkbox"/> Failed Auto Activations	<input type="checkbox"/> Failed Manual Activations
<input type="checkbox"/> Successful Auto Activations	<input type="checkbox"/> Successful Manual Activations
<input type="checkbox"/> Successful Auto Reactivations	<input type="checkbox"/> Successful Manual Reactivations
<input type="checkbox"/> Failed Online Key Validation	<input type="checkbox"/> Successful Online Key Validation
<input type="checkbox"/> Failed License Modification	<input type="checkbox"/> Successful License Modification
<input type="checkbox"/> Failed Auto Deactivation	<input type="checkbox"/> Successful Auto Deactivation
<input type="checkbox"/> Failed Manual Deactivation	<input type="checkbox"/> Successful Manual Deactivation
<input type="checkbox"/> Failed Generation	<input type="checkbox"/> Successful Generation
<input type="checkbox"/> Failed License Availability Check	

**License Expire Notifications**

☐ Days to License Expire:  ☐ After Expired

☐ Days to Maintenance Expire:  ☐ After Maintenance Expired

OK Cancel

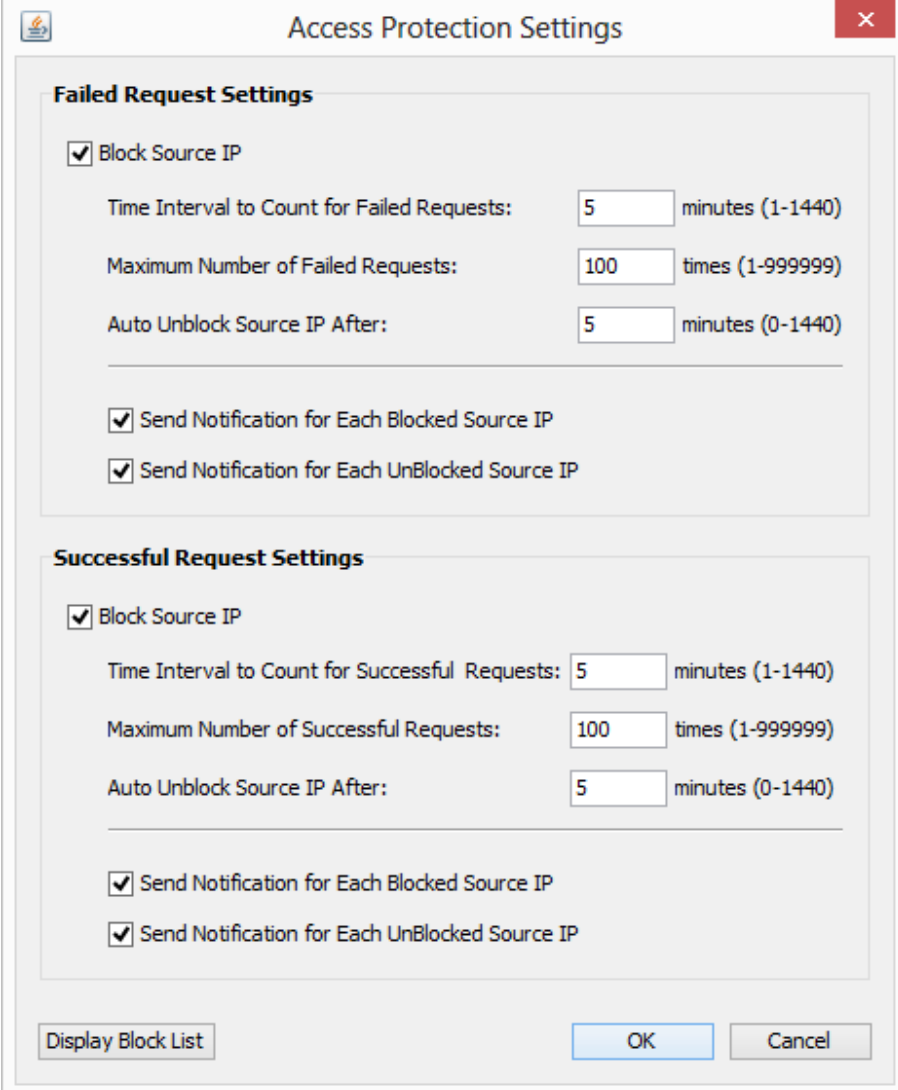
For each selected successful and/or failed generation and/or activation, server sends notification mails to addresses defined in *To Address* field. Multiple e-mail address can be specified separated with a semicolon.

Only one license expire notification is sent for each license if notifications enabled. Notifications are sent either after license or maintenance expired or defined days left to expiration. It can connect any SMTP server to deliver mails, and supports SMTP authentication and TLS.

Access protection settings can be defined for server so it can block requesting IPs if defined limits exceeded. Settings and limits for failed requests and successful requests are defined separately. "Time Interval to Count for Failed/Successful Requests" defines time interval to save all failed and successful requests to database. If a single IP address access count exceeds the maximum number of failed/successful access count, then the source IP is added to the block list and all new requests are blocked for a duration of value defined in "Auto Unblock Source IP After" field. If the defined value in "Auto Unblock Source IP After" is zero, then the blocked IP address is never removed from the list automatically, but it can be removed manually. The current block list can be viewed with "Display Block List" button.

If e-mail server and recipients are defined in notification settings window, then all block and unblock notifications can be send to defined recipients.

In the screenshot below, if server receives 100 failed or successful requests from same IP address in last 5 minutes, it blocks IP address for 5 minutes. After 5 minutes, it removes IP address from block list.



The image shows a Windows-style dialog box titled "Access Protection Settings". It has a standard icon in the top-left corner and a red close button (X) in the top-right corner. The dialog is divided into two main sections: "Failed Request Settings" and "Successful Request Settings". Each section contains a "Block Source IP" checkbox, which is checked in both. Below each checkbox are three input fields: "Time Interval to Count for Failed/Successful Requests" (set to 5 minutes), "Maximum Number of Failed/Successful Requests" (set to 100 times), and "Auto Unblock Source IP After" (set to 5 minutes). At the bottom of each section are two more checkboxes: "Send Notification for Each Blocked Source IP" and "Send Notification for Each UnBlocked Source IP", both of which are also checked. At the bottom of the dialog, there is a "Display Block List" button on the left, and "OK" and "Cancel" buttons on the right.

**Access Protection Settings**

**Failed Request Settings**

- ☒ Block Source IP
  - Time Interval to Count for Failed Requests: 5 minutes (1-1440)
  - Maximum Number of Failed Requests: 100 times (1-999999)
  - Auto Unblock Source IP After: 5 minutes (0-1440)
- ☒ Send Notification for Each Blocked Source IP
- ☒ Send Notification for Each UnBlocked Source IP

**Successful Request Settings**

- ☒ Block Source IP
  - Time Interval to Count for Successful Requests: 5 minutes (1-1440)
  - Maximum Number of Successful Requests: 100 times (1-999999)
  - Auto Unblock Source IP After: 5 minutes (0-1440)
- ☒ Send Notification for Each Blocked Source IP
- ☒ Send Notification for Each UnBlocked Source IP

Display Block List      OK      Cancel

## Error and Access Logging

Server stores access and error log files in a folder names as **logs** in *ProgramData* folder (*C:\ProgramData\ALGAS\logs*) on Windows and in installation directory on Linux. It keeps two log files: *license4j-server.log* file includes server messages and errors; *license4j-access.log* file includes access messages.

## Server License

One license is required for each computer running License4J Auto License Generation and Activation Server. Download package includes a 30 days valid trial license, and it activates trial license on first run.

When a license is purchased, new license key should be set in server settings window. Given license key will be auto activated on first run.

End of document.