

Plugins

Translate: EN



"Plugins" section is intended for controlling plugins - the server functionality extension modules in the form of DLL-files. Plugins allocate in /plugins server folder, which the server views at each start for creation of the list of existing plugins. For the plugins located in the list, it is possible to change priority by "Move Up" and "Move Down" commands the context menu.

Administrator

MetaTrader server

Common

IP Access List

Data Centers

Time

Holidays

Symbols

Securities

Groups

Managers

Data Feeds

Backup

LiveUpdate

Synchronization

Plugins

Accounts

Orders

Charts

Ticks

Journal

Name	Version	Copyright	File
LogAnalyser Settings	1.00	MetaQuotes Software Corp.	LogAnalyser.dll
TopTraders Settings		ware Corp.	TopTraders.dll

⚙️ Edit... Enter

⬆️ Move Up

⬇️ Move Down

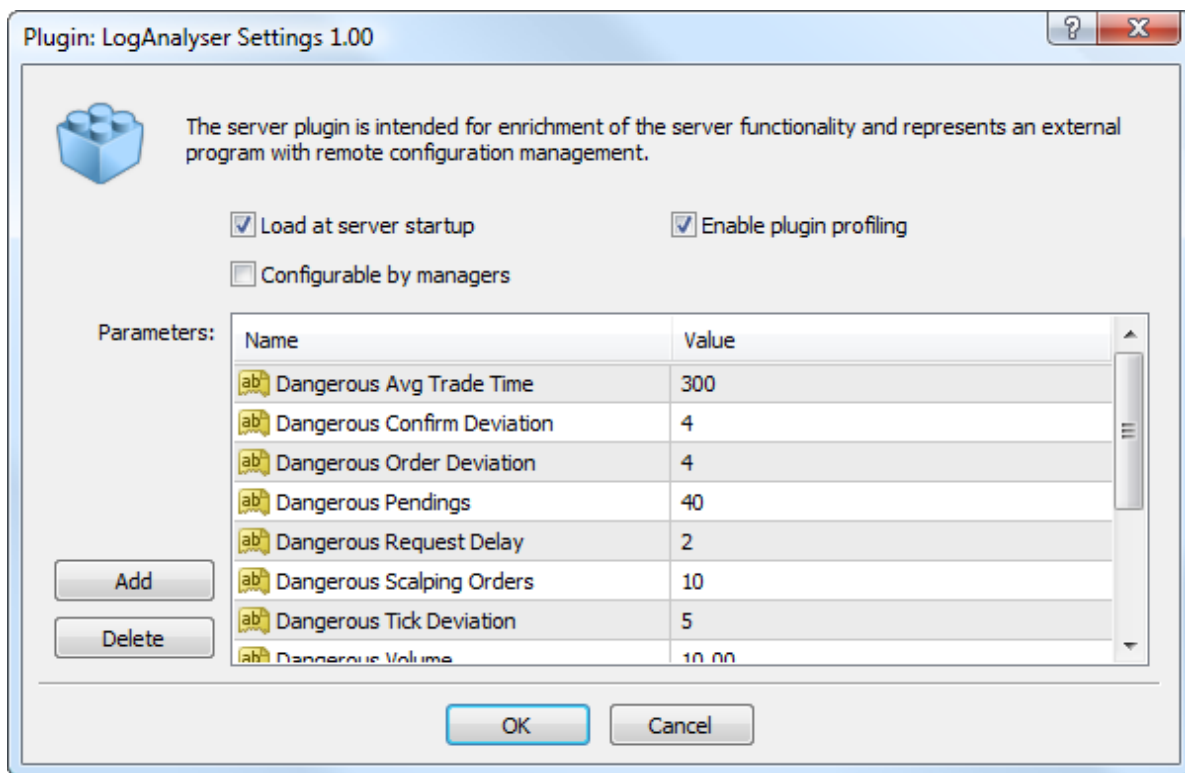
🔍 Find... Ctrl+F

🔍 Find Next F3

✓ Auto Arrange A

Plugin settings

To change plugin settings, it is necessary to execute the "Edit – Edit" menu command or the same command of the context menu. The window with detailed settings of the plugin will appear:



Load at server startup — load the plugin at server startup;

Configurable by managers — plugin can be configured in the manager terminal if the manager account has the appropriate right.

Enable plugin profiling — if [profiling](#) is enabled, the server collects statistics on the plugin operation, which can be used to evaluate the plugin performance and identify possible problems. Do not keep this parameter permanently enabled, because it affects the performance of the plugin and the trading server.

The plugin, as well as the profiling mode, is enabled/disabled after server restart.
Changes in priority of plugins (change of position) take effect after server restarting.

Each plugin can contain a set of some parameters serving for adjustment of its operation. Current parameters are displayed in "Parameters:" of plugin. For changing value or name of the parameter, it is necessary to select a necessary field in the list using doubleclick. After changing the edited field, it is necessary to press Enter key. To add the new parameter, it is necessary to press "Add" button. To delete the selected parameter, it is necessary to press "Delete" button.

Check of Plugins on Loading

When loading plugins, the trade server performs several checks. In case it detects a problem, the corresponding message is written to the server [journal](#):

The server measures the number of threads in the system before and after loading a plugin. In case the number of threads increases by more than 50, it will write the corresponding message in the journal.

In case a plugin include the [debug section](#), the corresponding message will be added to the journal.

It checks a digital signature of plugins. If there is no signature, the corresponding message will be added to the journal. If there is a signature, the journal message will display the name of an organization, who has signed the plugin.

If a plugin uses Java or .NET, the journal message will recommend not using this plugin. Such recommendation is due to using Java or .NET in plugins increases the loading of the trade server.

After loading a plugin dll, the server checks, whether exception handlers have been substituted (SetUnhandledExceptionFilter, _set_new_mode, _set_new_handler, _set_invalid_parameter_handler). In case it finds a substitution of handlers, the following message will be written to the journal: "it is recommended to remove handler replacement in the plugin as it negatively impacts the server's self-test system". This message may also appear due to compiling a plugin using Microsoft Visual Studio 6.0.

Profiling

Profiling allows evaluating the performance of the plugin by measuring the time and number of plugin hook calls. Measurement is performed for each hook available in the plugin.

To start profiling, enable the corresponding option in [plugin settings](#) and restart the trading server. After that, the server will begin to collect the plugin operation statistics and will output it to the [Journal](#). Statistical data is output once a minute after the monitoring data. To request logs, use the "API" keyword.

The profiling mode is only recommended for the plugin testing time, because it slows down the plugin operation.

Statistics is the cumulative data on all hooks of all plugins that were called during the trading server operation:

```
22:10:01.382      API      : 'Routing.dll' - MtSrvHistoryTickApply profile, to
22:10:01.384      API      : 'Routing.dll' - MtSrvTradeTransaction profile, to
```

The following information is displayed for all hooks:

total time — the total time spent in the hook. Together with "count", the value allows identifying an excessive amount of hook calls.

count — the total number of hook calls during the plugin operation time.

max — the maximum time spent in the hook. The value allows tracking rare, but large delays in performance.

calls — the number of current calls of the hook, which the plugin has not yet exited. The value allows detecting the full freeze of the hook.

If the hook call time exceeds 50 milliseconds, the following warning is added to the journal (no more than once every 5 seconds):

```
22:09:49.860      API      : 'Routing.dll' - MtSrvTradeTransaction profile, ca
```

In this case, it is strongly recommended to disable the plugin and optimize it. Otherwise, the performance of the trading server can be significantly reduced or completely hang.