

# IBController User Guide

---

## Introduction

### Overview of IBController

---

IBController enables Interactive Brokers' Trader Workstation (TWS) and Gateway to be run in 'hands-free' mode, so that a user need not be present. This makes possible the deployment of unattended automated trading systems.

IBController loads TWS or the Gateway and then 'listens' for various events (such as the display of dialogs) that would normally require user intervention, and then automatically takes appropriate action on the user's behalf. For example, as well as automating the TWS and Gateway login by filling the login dialog with your credentials, it also deals with TWS's autologoff dialog so that it can keep TWS running continuously.

Here are some of the things IBController does for you:

- starts TWS or the Gateway
- logs you into TWS or Gateway
- clicks the YES button if the "Accept incoming connection?" dialog is displayed.
- clicks the Close button if the Tip of the Day dialog appears.
- dismisses the dialog that warns of a new TWS version upon startup.
- responds to TWS's 'Exit Session Setting' to prevent autologoff, enabling TWS to be kept running indefinitely.
- shuts downs TWS tidily at a specified day of the week and time.

IBController also responds to certain commands sent to it by another program, for example to tell TWS/Gateway to shut itself down cleanly.

# Scope of this User Guide

---

This User Guide is intended to help you get started with IBController. It applies to IBController releases 2.12.0 and later.

The equivalent information for earlier releases can be found in the `readme.txt` file included in the relevant release's ZIP file. You can also view these readme files online. For example, the [version 2.11.0 readme.txt](#) provides detailed instructions on configuring that version.

Note that in the remainder of this document, 'Unix' is used to refer to all Unix-derived operating systems, including Linux and OS/X.

## Getting Started

### Checklist

---

Here is a summary of the steps you need to perform to get IBController up and running properly.

1. Ensure Java 7 is installed (see *Java 7 Runtime* in the *Prerequisites* section).
2. Install Interactive Brokers Trader Workstation (see *Interactive Brokers Trader Workstation* in the *Prerequisites* section)
3. Download the IBController distribution ZIP file (see the *Where to get IBController* section).
4. Install IBController (see the *Installation* section).
5. Create an encrypted folder called `IBController` in your personal filestore (see *Protecting the Password* in the *Password Security* section).
6. Copy the configuration file (called `IBController.ini`) from the IBController installation folder to the encrypted folder created in step 5.
7. At this stage, everything is set up to run IBController with its default settings, which will start TWS and log it into the IB demo user. It is

worthwhile doing this just to check that everything works before customising it to suit your needs. To do this, run the relevant shell script (IBControllerStart.bat on Windows, IBControllerStart.sh on Unix) from the IBController installation folder. If everything is satisfactory, you can shut down IBController in the usual way.

8. Edit the configuration file ( `IBController.ini` ) in the encrypted `IBController` folder using a text editor such as Notepad. See *Configuring IBController* for further information.
9. If you did not install TWS and IBController in their default locations, and store the configuration file in the recommended location, you will have to edit the shell scripts in the IBController installation folder accordingly. They contain comments that will help you do this correctly.
10. If you intend to run API programs to connect with TWS, you will need to manually edit the API settings in TWS's Global Configuration Dialog.

## Prerequisites

---

This section details the other software that is needed to run IBController.

Note that some Unix distributions may provide packages that can automatically install and configure everything needed to run IBController, for example the Arch Linux [ib-controller](#) package. This subject is beyond the scope of this document.

### Java 7 Runtime

Both IBController and TWS/Gateway are Java programs, and therefore the Java Runtime version 7 or later needs to be installed. If you aren't sure whether the Java Runtime is already installed, or what version is installed, issue the following command from a command prompt:

```
java -version
```

For Windows, you can download and install the Java Runtime from its [official download site](#).

For Unix you can install Java using your package manager.

### Interactive Brokers Trader Workstation

Before running IBController, you will need to download and install the standalone version of Trader Workstation from the [Interactive Brokers](#) website. Note that this installation includes the code for both TWS and the Gateway.

Note that you do not need an IB account to try out IBController, as you can use the IB demo account (username `edemo` , password `demouser` ).

## Where to get IBController

---

IBController is officially distributed as a ZIP file containing the compiled program and some additional files, detailed below.

The ZIP file for the latest version should be downloaded from [Github](#). Earlier versions can also be downloaded from the same place if need be.

The distribution ZIP file contains:

- [License](#) text
- Compiled JAR (named similar to `IBController.jar` )
- Sample configuration file (named similar to `IBController.ini` )
- Sample TWS launch script for Windows (named similar to `IBControllerStart.bat` )
- Sample Gateway launch script for Windows (named similar to `IBControllerGatewayStart.bat` )
- Sample TWS launch script for Unix (named similar to `IBControllerStart.sh` )
- Sample Gateway launch script for Unix (named similar to `IBControllerGatewayStart.sh` )
- Sample Windows Task Scheduler file (named similar to `Start TWS Live (daily).xml` )

Source code and build scripts are not included in the distribution ZIPs, as they are freely available from the [IBController project page](#) on Github.

## Installation

---

Installing IBController is just a matter of extracting the contents of the downloaded ZIP file to wherever you want to install it. You will make things easiest for yourself if you use the the locations described in 'Default Paths' below, because that will minimise customising the configuration file and the shell scripts.

On Windows:

- create the folder where you want to install IBController, if it doesn't already exist. As noted above (see Default Paths) this is normally `C:\IBController`

but it can be anywhere you like

- locate the downloaded ZIP file using File Explorer (Windows Explorer on Windows 7 and earlier). Windows treats ZIP files like an ordinary folder, so you can see its contents the same way as any other folder
- select the files and drag them into your installation folder

On Unix:

- use a command similar to this:

```
cd /opt
sudo unzip ~/Downloads/IBControllerV2.12.0.zip -d /opt/IBController
```

## Default Paths

Several sample files ship in each IBController release. The sample files (and these instructions) assume the default paths shown in the table below (where `<username>` represents your operating system user name, not your IB login id).

If you store any of these items in other locations, you will need to edit these sample files to reflect this.

Platform	Item	Path
Windows	IB TWS program files	C:\Jts
	IBController program files	C:\IBController
	IBController.ini	%HOMEPATH%\Documents\IBController
Unix	IB TWS program files	/opt/IBJts
	IBController program files	/opt/IBController
	ibcontroller.ini	/home/<username>/IBController

Note that installing IBController and/or TWS from a Unix package manager may not use these paths. Consult your Linux package instructions for file locations.

## Password Security

---

To login to TWS or IB Gateway, IBController needs to know your Interactive Brokers username and password. You should very carefully secure your IB account username and password to prevent unauthorised use by third parties. This section gives you guidance on how to achieve this.

The username and password are given to IBController in one of two ways:

- via the configuration `.ini` file: this is the preferred method because the configuration file can be protected by the operating system
- via the command line parameters when IBController is started: this method is strongly deprecated because command line information associated with a process is easily available outside the process (for example via Task Manager on Windows)

## Protecting the Configuration File

To protect this sensitive information, the configuration file needs to be stored in a location where it will not be accessible to other users of the computer. The simplest way to achieve this is to store it within your personal filestore:

- on Windows this is your `Documents` folder (which is normally actually located at `C:\Users\<username>\Documents` )
- on Unix it is the `/Home/<username>` directory.

You are advised to place the file in its own `IBController` folder within this location.

You should also consider encrypting the folder containing the configuration file. This will prevent another user with administrator privileges gaining access to the contents: even if they (ab)use their administrator privileges to give themselves access to the file, its contents will not be decrypted because they are not the user that encrypted it.

To encrypt the folder on Windows:

- right click the folder and select `Properties`
- click the `Advanced` button on the `General` tab
- set the checkbox labelled `Encrypt contents to secure data`
- finally, click the `OK` buttons to apply the changes.

Encrypting a folder on Unix is more involved, and you should refer to the documentation for your distribution.

## Encrypting the Password

When the password is included in the configuration file, IBController allows it

to be in an encrypted form so that casual observers cannot see the actual password, for example while you are editing the configuration file. Note however that the encryption is very simple and easily reversed by anyone who knows the encryption technique, so it does not remove the need to store the configuration file securely.

To encrypt your password, run the following command from the IBController directory:

```
java -cp IBController.jar ibcontroller.IBController encrypt <your-password>
```

The program output will include your encrypted password, which can then be included in your IBController configuration `.ini` file.

You should also clear your shell history if you entered the above command. In most cases this is achieved by pressing ALT+F7 (Windows users) or typing `history -c` (Linux user), although you should check it worked by pressing the up arrow afterwards.

## Configuring IBController

---

IBController must be supplied with a configuration file. A specimen file called `IBController.ini` is included in the distribution ZIP. You will need to edit this file to include your IB username and password, and to ensure that IBController behaves in the way that best suits your needs.

You should copy the supplied file from the IBController installation folder into the secure location described above before editing it, so that you have a clean copy to revert to if need be.

The sample `.ini` file contains detailed comments on the meaning of each configuration property. Many of these have sensible defaults, or are only needed in special situations, so to help you get started quickly, here is a list of the settings that you are most likely to need to change:

Setting	Notes
IbLoginID	You must set this to your IB username
PasswordEncrypted	You must set this to <code>yes</code> if you have encrypted your password
IbPassword	You must set this to your IB password (possibly encrypted)

<code>IbDir</code>	You will need to set this if you are using Unix
<code>AcceptIncomingConnectionAction</code>	It is safest to set this to <code>reject</code> and to explicitly configure TWS to specify which IP addresses are allowed to connect to the API
<code>IbAutoClosedown</code>	Set this to <code>no</code> to prevent TWS's daily auto closedown
<code>ClosedownAt</code>	Set this if you want to keep TWS running all week

There are two ways that IBController can locate your edited `.ini` file.

- the simplest way is to tell it where to find the file in the command that starts IBController. If you do this, you can give the configuration file any name you like. This is the recommended approach. You will need to edit the execution shell script (details below) if you change the filename from `IBController.ini`, or if you store it somewhere other than the default location
- if you do not specify a configuration file name, IBController will expect to find a file named `IBController.<username>.ini` in the current working directory. In this case, `<username>` is your username on your computer (not your IB account username). This method is deprecated, because it is likely to result in the `.ini` file being in an insecure location.

## Starting IBController

The normal way to start IBController is by use of a shell script. These can be identified by the `.bat` (Windows) or `.sh` (Unix) extensions. Sample scripts to start TWS and Gateway are included in the distribution ZIP.

Windows users can execute a shell script in a number of ways, including:

- Double-click the filename in Windows Explorer
- Create a shortcut to it on your Start menu, desktop or taskbar
- Create a scheduled task to run it automatically at the required times (see below for more information about using scheduled tasks)

If you used the default locations to install IBController and TWS, and to store your `IBController.ini` file, you should not need to edit the shell scripts. If you do need to change them, they are commented to help you.



# Other Topics

## Scheduled Tasks (Windows)

---

On Windows you can start IBController automatically using a Scheduled Task.

If you do this, you must make sure that the machine is already logged on before the scheduled task runs. Otherwise the task will still run, but you won't be able to see and interact with TWS, even if you subsequently log on.

Remember also to change the task settings to prevent Windows automatically ending it after a certain time.

Also you can use the `IbAutoClosedown=no` setting in the IBController configuration file to disable TWS's autologoff feature, and the `ClosedownAt=` setting to specify when IBController will shut down TWS.

In this way you can start IBController automatically on Sunday evening or Monday morning, keep it running all week and then close down tidily on Friday evening or Saturday morning.

The Windows Task Scheduler has many powerful features, and some of these can be used to provide even better control. For example, you can run the task every 5 minutes during the week so that if TWS crashes, it will automatically be restarted within 5 minutes. If you also set up your computer to log on automatically when it starts (information about this is easily available on the internet), this will also restart TWS after a power outage (but make sure you understand the security implications of autologon to Windows).

A sample scheduled task is included in the IBController distribution ZIP, called `Start TWS Live (daily).xml`. You can import this into your Task Scheduler if you are running Windows 7 or later. After importing it, you will need to enable it and change the user account it runs under. This task starts TWS daily at 05:55, and assumes that TWS is set to auto-logoff at 05:52, so the IBController configuration file must include `IbAutoClosedown=yes`.

## Multiple IBController Instances

---

You may want to run more than one instance of TWS or the Gateway on the same computer, perhaps simultaneously. Here are some reasons you might want to do this:

- you want to run both your live and paper-trading IB accounts. This is especially true if you want to get market data from both accounts, as IB will only allow this if both TWS instances are on the same computer (unless you don't try to run them at the same time)
- you have multiple logins for your live IB account, and want to run TWS for both, perhaps at the same time
- you trade on behalf of others, perhaps your family, friends or clients, who each have their own accounts, but you want to run TWS instances for all these accounts on one powerful computer
- you want to trade in different regions at different times
- you want to test a new version of TWS in your paper trading account at the same time as using your live account in a previous version

When TWS runs, it stores a large number of settings in a folder structure (these settings may also be stored in IB's servers, but this is not a useful option if you want to use multiple TWS instances). By default, TWS stores this settings folder structure in the TWS installation folder.

By using the `IbDir` setting in the `IBController` configuration file, you can tell TWS to store its settings wherever you like. So to have multiple `IBController` instances, you need to create a separate configuration file for each instance with a different setting for `IbDir`. Note that you do not need to copy the TWS .jar files themselves into these folders - you can load TWS from the same installation folder for each instance.

Because you now have different configuration files, you also need different scripts to run each instance (or you could have a single script and pass the configuration file details as a parameter).

As a concrete example, let's take the first scenario described above: you want to run both your live and paper trading accounts. So:

- install TWS into the default location ( `C:\Jts` on Windows)
- create two new folders `C:\JtsLive` and `C:\JtsPaper` to store the settings
- create two `IBController` configuration files called `IBControllerLive.ini` and `IBControllerPaper.ini`
- set the `IbDir` option in them to point to the relevant folder, ie `IbDir=C:\JtsLive` and `IbDir=C:\JtsPaper`, and set the `IbLoginId` and `IbPassword` to the live or paper account values as appropriate
- create two start scripts (by copying `IBControllerStart.bat`) called `IBControllerStartLive.bat` and `IBControllerStartPaper.bat`

- change the `set IBCINI=...` line in each script file to refer to the relevant configuration file
- now you can run the new scripts, and each will start a separate instance of TWS connected to a different account, with its settings stored in separate folders.

## Using different TWS versions simultaneously

To use more than one version of TWS (for example for testing a new version with your paper-trading account while also using a previous version for your live account), you need to be a bit careful about installing TWS.

The suggested approach is to install each TWS version into its own folder (for example install TWS 946 into `C:\Jts946`). Then have separate settings folders for the different modes you want to operate, for example `C:\JtsSettingsLive` and `C:\JtsSettingsTest`. Next adjust the start scripts and configuration files to reflect these locations. You can now run the stable and test versions of TWS simultaneously. When you are happy that the test version works correctly, you can easily start using it for your live account by adjusting the `set TWSDIR=...` line in the start script for the live account.

Note that when you install more than one version of TWS in this way, only the last one installed can be uninstalled via the Control Panel's Program and Features applet. Other versions can be uninstalled by running the `UNWISE.EXE` program in the relevant TWS installation folder.

## Any Questions?

---

Lots of people use IBController and are happy to assist you! Please see the [support information](#) for details.