

# Windows System Commands Plugin

As the name suggests, the Windows System Commands Plugin is used to directly run Windows system commands that have been entered by the user as a query. This is done by parsing the entry and validating the command, followed by executing it.

The user can change the behavior of the plugin (language, confirmation dialog, ...) with optional plugin settings.

 Image of System Commands plugin

Available commands:

- Shutdown
- Restart
- Sign Out
- Lock
- Sleep
- Hibernate
- Empty Recycle Bin
- UEFI Firmware Settings (Only available on systems, that boot in UEFI mode.)
- IP / MAC / Address => Show informations about network connections.

## Optional plugin settings

- We have the following settings that the user can configure to change the behavior of the plugin:

Key	Default value	Name/Description
<code>ConfirmSystemCommands</code>	<code>false</code>	Show a dialog to confirm system commands
<code>LocalizeSystemCommands</code>	<code>true</code>	Use localized system commands instead of English ones
<code>ReduceNetworkResultScore</code>	<code>true</code>	Reduce the priority of 'IP' and 'MAC' results to improve the order in the global results

- The optional plugin settings are implemented via the [ISettingProvider](#) interface from `Wox.Plugin` project. All available settings for the plugin are defined in the [Main](#) class of the plugin.

## Technical details

### [Main](#)

- Tries to parse the user input and returns a specific Windows system command by using a [Result](#) list.
- While parsing, the plugin uses [FuzzyMatch](#) to get characters matching a result in the list.

### [Commands.cs](#)

- The [Commands](#) class contains the definition of all available commands/results.

### [ResultHelper.cs](#)

- The [ResultHelper](#) class contains methods for working with the results and some of the result features (tool tip, copy to clipboard, execute command).

### [NetworkConnectionProperties.cs](#)

- The [NetworkConnectionProperties](#) class contains methods to get the properties of a network interface/connection.
- An instance of this class collects/provides all required informations about one connection/adapter.

### [SystemPluginContext.cs](#)

- An instance of the class [SystemPluginContext](#) contains/defines the context data of a system plugin result. We select the context menu based on the defined properties.
- It is used for the `ContextData` property of the [Wox.Plugin.Result](#) .

## UEFI command

- The UEFI command is only available on systems, that boot in UEFI mode.
- This is validated by checking the result of the method [GetSystemFirmwareType](#) , which uses the native method [GetFirmwareType](#) in `kernel32.dll` .

## Search

### Score

- [CalculateSearchScore](#) A match found near the beginning of a string is scored more than a match found near the end. A match is scored more if the characters in the patterns are closer to each other, while the score is lower if they are more spread out.
- For network results (IP address and MAC address) the score is reduced by 25 percent.

### Network results on global queries

- The network results (IP and MAC address) are only shown on global queries, if the search term starts with either IP, MAC or Address. (We compare case-insensitive.)

### Returning results

We return the results in two steps:

1. All results which we can create very fast like shutdown or logoff via [Main.Query\(Query\\_query\)](#) .
2. All results which need some time to create like the network results (IP, MAC) via [Main.Query\(Query\\_query, bool delayedExecution\)](#) .

## Unit Tests

We have a [Unit Test project](#) that executes various test to ensure that the plugin works as expected.

### [ImageTests.cs](#)

- The [ImageTests.cs](#) class contains tests to validate that each result shows the expected and correct image.

### [QueryTests.cs](#)

- The `QueryTests.cs` class contains tests to validate that the user gets the correct results when searching.