# Maelstrom CD-Key Tester

1. Introduction

2. The Configuration

3. Dealing With Configuration Issues

4. Setting Up Key Testing

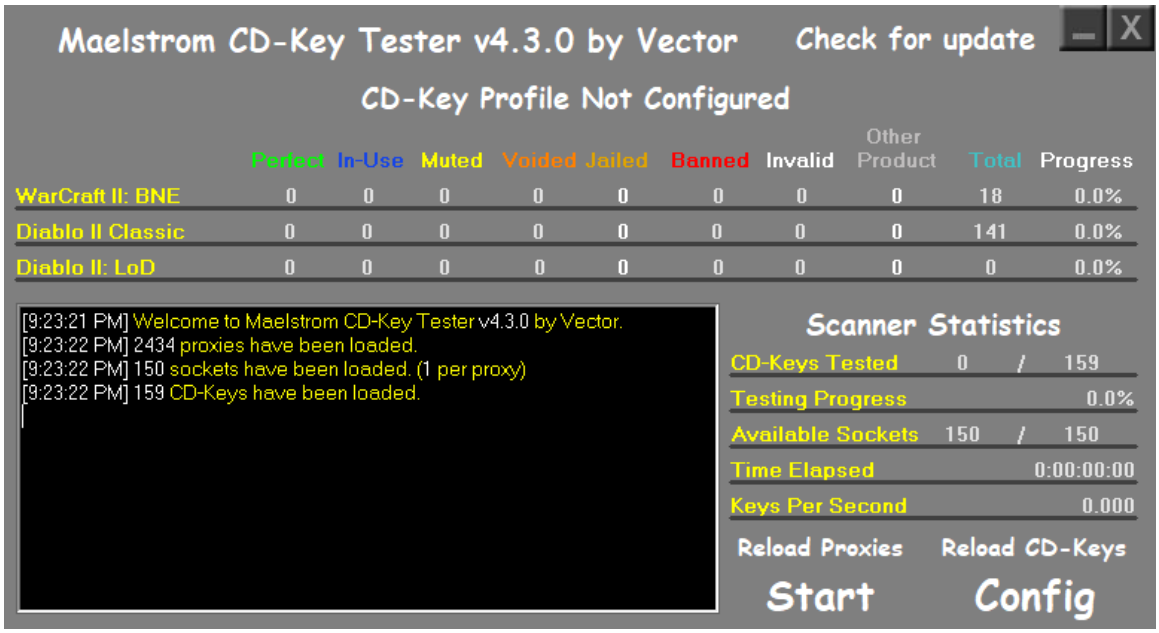
5. Expansion Key Testing

6. CD-Key Profiles

7. Tips and Tricks

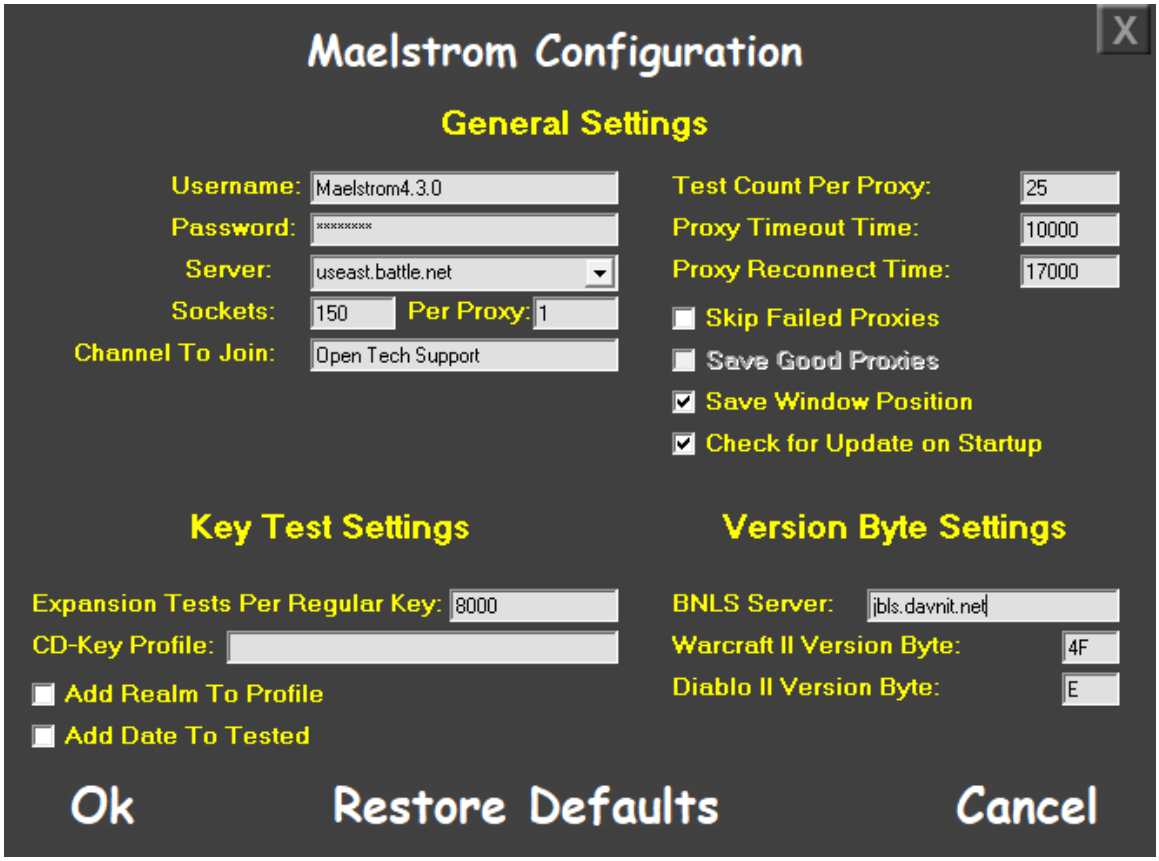
8. Special Thanks

## 1. Introduction




Maelstrom CD-Key tester is a program that is used to test many keys at once. It works with SOCKS4, SOCKS5 and HTTP proxies. It can even test expansion keys. There are also a number of different settings that can be used to customize Maelstrom.

# 2. The Configuration




The configuration is where you can fine tune your key testing environment to your liking. The default values work well but you might want something different. The "Restore Defaults" button will restore all default settings to the configuration. There is also a maximum setting for the following values: sockets (32767), sockets per proxy (8), expansion tests per reglar key (32767), test count per proxy (32767), reconnect time (32767), check failure (32767).

Some of the values are self-explanatory but for the rest they are explained below:

"Proxy Timeout Time" (in milliseconds) specifies how long to wait to connect to a proxy before the connection times out.

"Proxy Reconnect Time" (in milliseconds) specifies how long after Maelstrom tests a key until it

reconnects to the proxy.

The "Test Count Per Proxy" setting specifies how many tests on a single socket connected to a proxy is done before that proxy is swapped out for another proxy.

The "Expansion Tests Per Regular Key" setting specifies how many expansion keys can be tested on a single regular key before that key is swapped out for another one.

The "Add Date To Tested" setting will save the tested keys to a folder in the format "mmmm  d, yyyy" (August 1, 2016 as example) allowing you to save tested keys by date. This setting can work alongside CD-Key Profiles.

The "Skip Failed Proxies" setting (enabled by default) will skip over proxies that do not respond. On each failure of the proxy it will be marked for a total amount of times as much as the "Sockets Per Proxy" setting before that proxy is removed from the list. Maelstrom will no longer try to connect to that proxy until the proxy list is reloaded.

The "Save Good Proxies" setting will save all unused and good proxies back to the various proxy files when Maelstrom is closed. Requires the "Skip Failed Proxies" setting to be enabled first.

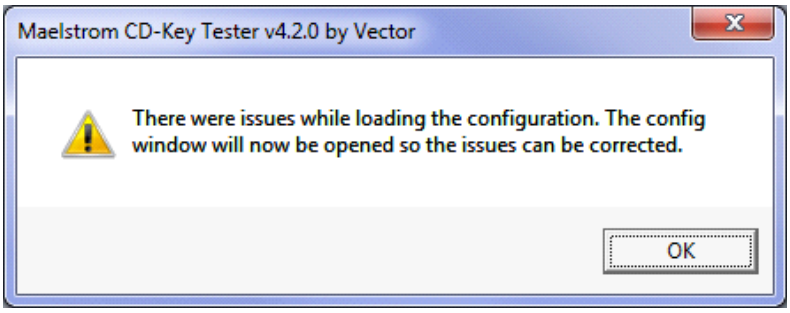
The "BNLS Server" setting specifies a BNLS server to update version bytes from. The version bytes below this setting specify the version of the products being tested. If Maelstrom tries to use an invalid version byte it will try to update the version byte from the BNLS server.

The "CD-Key Profile" is used along with "Add Realm To Profile" to specify a custom output folder for saved keys. More is explained in section 6.

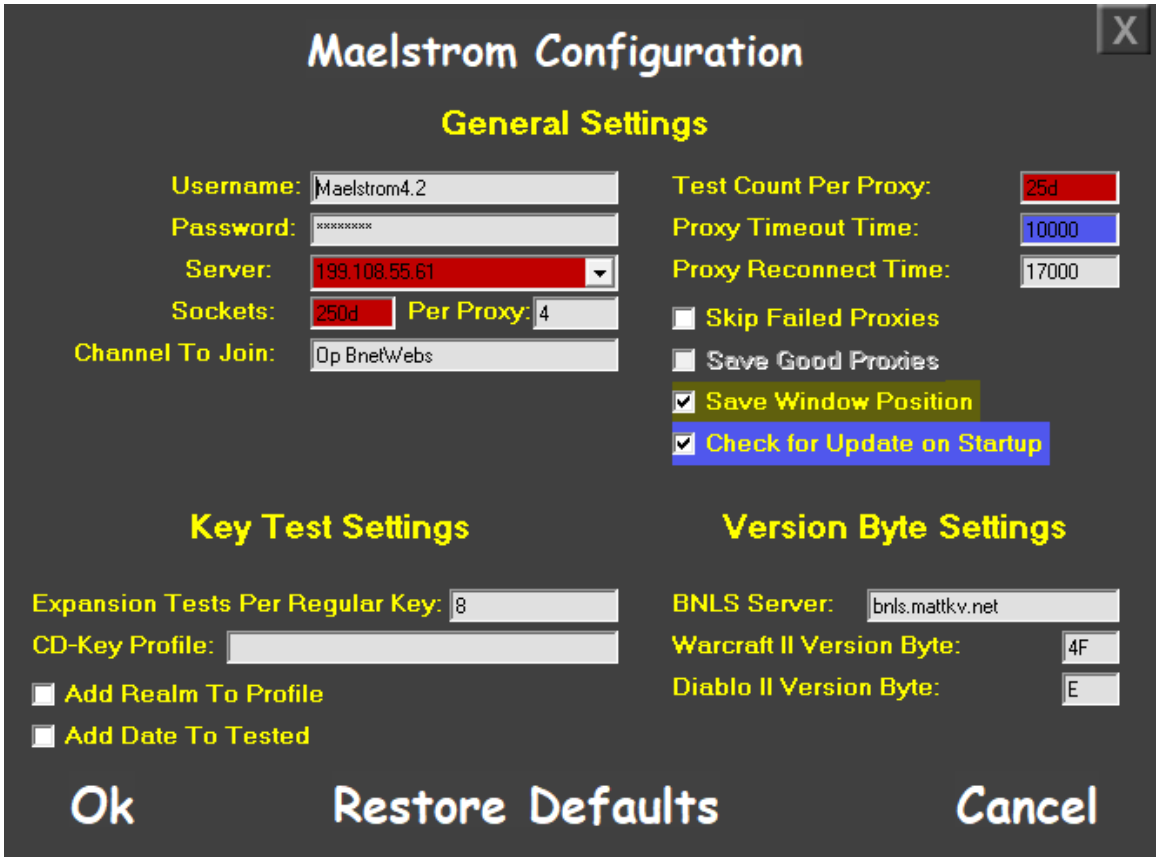
"Save Window Position" will save the position of the window after exiting and apply it once the program starts again.

"Check for Update on Startup" will check for updates when the program is started. This option is enabled by default.

# 3. Dealing  With Configuration Issues




In the event that Maelstrom loads with invalid settings in Config.ini or values aren't present the configuration window will load up with three specific color codes indicating the type of issue with each configuration setting that did not load correctly.




Settings in red indicate the setting was invalid, settings in light blue mean the setting was not present in the config and was filled in with a default value and settings in yellow mean the setting was invalid but it can be ignored as it was given a default value.

Anytime you modify the config with invalid values and then hit "OK" Maelstrom will also highlight any text boxes in red that have an invalid value. The configuration will not be saved until all errors in red are corrected.

# 4. Setting Up Key Testing

Since this program utilizes SOCKS4, SOCKS5 and HTTP proxies, SOCKS4.txt, SOCKS5.txt and

HTTP.txt have been provided for you to place proxies in. For keys, the folder "CD-Keys" is for you to place your keys in. You can drop in an entire folder structure into this folder or a file of keys and Maelstrom will read it all in when starting. This is useful if you'd like to re-test a previous list that Maelstrom had tested or would like to drop in several files to test. Simply drop everything into the "CD-Keys" folder and either load up Maelstrom or if Maelstrom is already loaded click the "Reload CD-Keys" label.

All files present in the "CD-Keys" folder with the exception of STAR.txt, W2BN.txt, D2DV.txt and D2XP.txt will be deleted. If these files are not present then they will be created automatically when Maelstrom loads the keys.

Since this program requires hashes you need the proper hashes in order to test keys for specific products. Check the "READ THIS.txt" file in the \Binaries\ folder for more information on where to get hashes. The lockdown libraries are already provided so you do not need to download those.

# 5. Expansion Key Testing

Maelstrom first needs to test non-expansion keys before it can start testing expansion keys. It can use perfect, muted, voided and jailed non-expansion keys to test expansion keys. It uses the "Expansion Tests Per Regular Key" setting in the configuration to specify how many expansion keys can be tested on a non-expansion key before that key is swapped out for a new key. The GUI will not properly show how many keys were tested until all non-expansion keys and all expansion keys are tested for a specific product.

# 6. CD-Key Profiles

The CD-Key Profiles feature allows you to specify a profile for the keys that are being tested. This effectively renames the output folder Maelstrom sends tested keys to. Without a profile

Maelstrom sends tested keys to the "Tested CD-Keys" folder. This is useful if you want to describe the type of keys you are testing. The profile name is permanent until changed in the configuration. If you wish to disable CD-Key profiles then leave the "CD-Key Profile" configuration setting blank.

The "Add Realm To Profile" configuration option allows the current realm to be attached to the profile. If your current profile is named "My Key List" and you are currently using an east server the final profile will be named "My Key List @ USEast". You could also test the same keys on all 4 realms. The profile will be updated according to each realm.

# 7. Tips and Tricks

Knowing how you can add keys to Maelstrom will really help save time in the future. You can add files as well as folders inside of the "CD-Keys" folder. Maelstrom will read all of the keys in each file, determining the product of each key so you don't even need to specifically put the right keys in the standard files.

Duplicate keys, unreadable keys, blank lines and invalid keys are ignored when Maelstrom loads keys. Maelstrom can also handle lines with keys that have extra content in them so you don't have to worry about removing extra information before using the key file with Maelstrom. An example of the kinds of things Maelstrom can read are as follows:

My keys:

-- 1234567890123 the key is the second line

1234567890123--- another line

----1234567890123---------

Dashes are always removed from a line when loading keys. Maelstrom will try to get a piece of

text with a standard key length. If unsuccessful it will get the biggest piece of text on the line and if it is greater than 26 characters will be treated as if it is 26 characters. Not every situation is handled so some keys may not be read in properly. This should not be the case the majority of the time.

# 8. Special Thanks

Sammy, Skye, James and Matt

- If it were not for these people asking me to write a key tester, I don't think Maelstrom would have ever existed. Thank you guys so much!

James (Anti-Liberal@USEast)

- For his ideas for this key tester, including the number of tests per proxy. He was there for the project's inception and has been a big help in more ways than one. His suggestions have made Maelstrom into something it would not have been without his support. Thanks James :P

Code Speak and others

- Thank you guys for motivating me to work on this project

Lynxy

- For providing a grey gradient used in earlier versions of Maelstrom

Travis (Simile@USEast)

- For the idea of adding dates to tested keys

Derek (Derek@USEast)

- For the idea of connecting on single proxies at a time but not connecting more sockets to a single proxy than the "Sockets Per Proxy" setting

Robert (evilmad666@Azeroth)

- For the extensive testing of Maelstrom 4.0, comparing it to 3.42, giving me ideas and giving me proxies I could test with. Thank you so much!

MaLi

- For helping me test and hunt down an issue related to proxy connections

Arana

- For his support over the years and helping me to test Maelstrom