



MS Windows Adaptive Agent For Windows 2000/2003

This guide serves as a compliment to the LoadMaster documentation, and is meant to get a new user up and running with basic functionality quickly. It is not a replacement for the full LoadMaster documentation. For concepts and issues not covered in this document, consult the *LoadMaster Installation and Configuration Guide*.

Table of Contents

What You Will Need.....	2
Overview.....	2
Setting up the Real Server.....	2
Configuration File	2
IIS Configuration.....	3
Registry	5
Repeat.....	6
Setting up the LoadMaster.....	6
Virtual Service Configuration	7



Copyright © 2000 - 2009 KEMP Technologies, Inc. All rights reserved.

KEMP Technologies, Inc. reserves all ownership rights for the LoadMaster product line including software and documentation.

The use of the LoadMaster Load Balancer is subject to the license agreement.

Information in this guide may be modified at any time without prior notice.

Company names used in examples are fictitious unless otherwise noted.

Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.

All other trademarks and registered trademarks are the property of their respective owners.

International copyright law protects this document. All rights are reserved. Copyright in the content, in the screens displaying the content, and their arrangement, is owned by KEMP Technologies, Inc. unless otherwise indicated. The content and data found on this document may not be redistributed, copied, sold, transferred, or modified, without the express written permission of KEMP Technologies, Inc. De-compilation, reverse engineering or disassembly of document is strictly prohibited.



What You Will Need

To utilize the LoadMaster server agent you will need the following:

- A LoadMaster installation, running at least LoadMaster version 3.1-57
- A Windows Server (2000 or 2003), running IIS (5.0 or later)
- The permissions needed to setup CGI applications under IIS

Overview

A KEMP LoadMaster determines how to best distribute load among available real servers by a variety of methods. These can include the availability of the servers (by doing health checking), as well as both active connections and connection rates to the real servers.

It may be beneficial however to also have information on the performance characteristics of the real servers themselves. This is possible through the “adaptive” load balancing method in the load balancer. In conjunction with an agent, the LoadMaster can distribute traffic based on the load on the actual server, which can include CPU load, disk load, and other performance metrics.

The LoadMaster Server Agent works by residing on a server and reporting back to the LoadMaster on how loaded the server is. The metrics used to determine the system's load are entirely configurable by the user.

The LoadMaster Agent simply reports a number, between 1 and 99, with 1 being the idlest and 99 being the busiest. The number is determined by the parameters specified in the configuration file. It can include only one performance metric (such as CPU utilization), or be a combination of up to eight different metrics (such as CPU utilization, memory utilization, disk I/O).

Setting up the Real Server

The first step is to choose where the agent will reside. The agent exists as a CGI program, and it must be somewhere within your webroot, available to the LoadMaster. The LoadMaster will retrieve the load value through an HTTP call, so it must exist in the webroot.

For instance, if your webroot is `C:\inetpub\wwwroot\site1\`, then you might place the agent in `C:\inetpub\wwwroot\site1\LMagent\`.

The two files required are `LMperfagent.exe` and `LMperfagent-config.txt`. They must be together in the same directory. There is also a file called `LMperfagent-config.xls`, which is a file you can edit under Microsoft Excel if you find that easier to work with. The file must be saved as a TXT file, however, the XLS file will not be



read.

Configuration File

If you look at the configuration file (either the txt file in a text editor, or the xls file in a spreadsheet editor), you'll see that each metric has its own line, and each line consists of five different fields.

The first field is the weight that the specified metric has on the overall performance value reported. The second field is the top range of the value reported by that metric. For example, if you were to measure system memory in terms of megabytes, and the system had 2 GB of internal memory, the maximum value would be 2048. If the memory utilization was 1024, then the value reported would be "50", since the memory is 50% utilized.

The third field is whether the value is an up or down value. A good example to illustrate the difference between an up or down metric would be a metric that reports memory. If the metric reports memory available, then it would be a "down" metric. As the metric value goes down, the value reported would go up. A system with 2048 MB of RAM that reports 0 memory available would be a fully utilized system, so it would report 101.

If instead the metric reported memory utilized, a value of 2048 MB of RAM would mean no RAM was being used, and the value reported would be 1. If the memory utilized shot up to 2048, then the value reported would be 101.

The fourth field is the hard limit value. Setting a value will cause the agent to override all other metrics, and ignore their weights, and report a 100% loaded condition if this threshold is reached. An example would be for CPU utilization. If you have CPU weighted to 50%, and you have disk I/O and pages per second each weighted at 25%, you can have a situation where disk I/O and pages per second are nearly at zero, but CPU hits 100%. In that situation, the agent would report 50% utilization to the LoadMaster. If you had a hard limit set for 100 for CPU, if CPU hit 100, then the agent would report 101.

And the fifth field is the metric itself. A list of available metrics can be found in the example configuration file. Additional information is available in the README file.

IIS Configuration

Once the files are in place, and the configuration is set (adjustments can be made later, of course), then IIS needs to be configured. First, open the IIS configuration manager, which can be brought up by going into Control Panel, Administrative Tools, and selecting IIS Manager.

Look for the "Web Service Extensions" menu, and click on "Add a new Web service extension..." (figure 1).



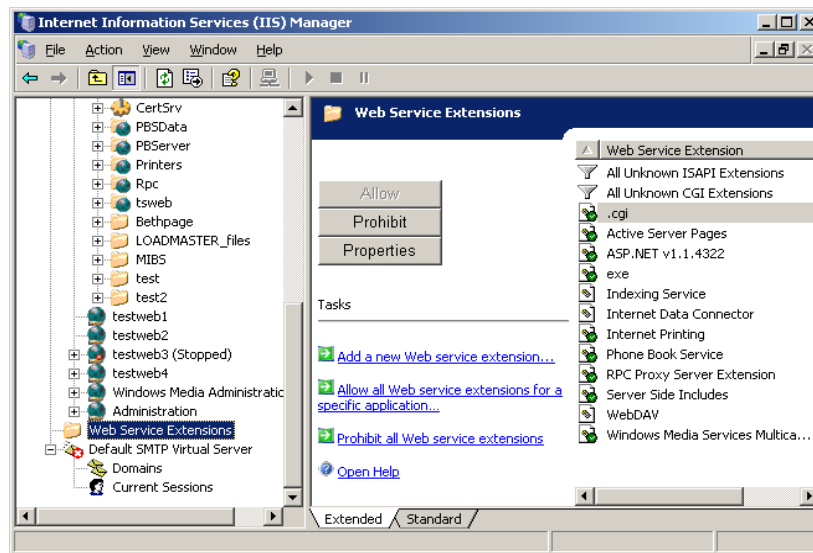


Figure 1: Web Services Extension

Under extension name, give “LMperfagent.exe”, then click “Add”. Browse for and select the LMperfagent.exe file (figure 2).

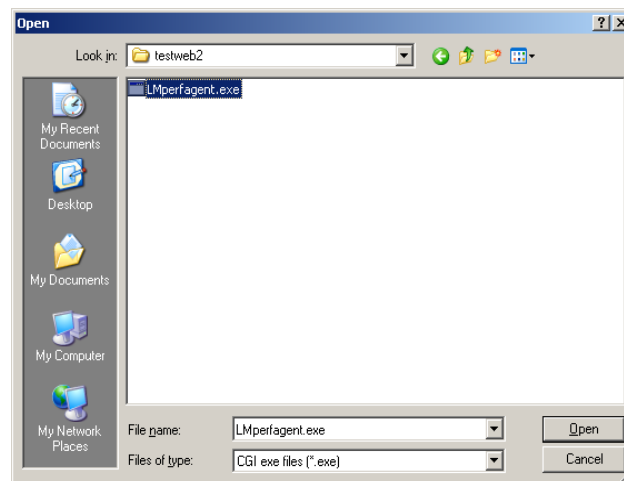


Figure 2: Adding Exectuable

Then go to “Web Sites” in the main window and select your site, and then select properties. Select the “Home Directory” tab, and change “Execute Permissions” near the bottom to “Scripts and Executables”. Then click on the “Configuration...” button, and then “add”.



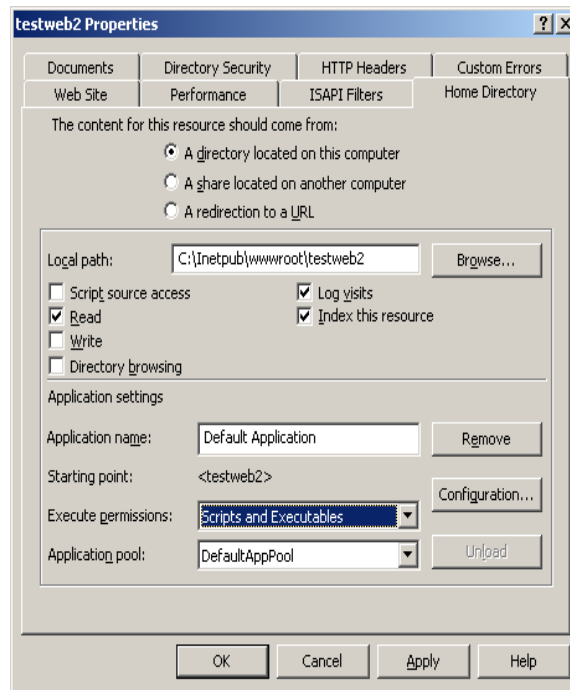


Figure 3: Home Directory Configuration

For executable, select the LMPerfagent.exe file, and put in “exe” under extension. Unselect “script engine”, and then click OK (figure 3).

Registry

There is one more step required, and that's to set the Windows registry to provide access to the perflib counters to the agent. Open up the Registry Editor, such as by going to the Start menu, selecting run, and typing in “regedit”.

From there, open up **HKEY_LOCAL_MACHINE**, then **SOFTWARE**, then **Microsoft**, then **Windows NT**, **CurrentVersion**. There you'll find Perflib. Select “Permissions” for Perflib.

Select “Add”, and the group you're adding is called “Everyone”. Once that's added, “Everyone” will have no read/write permissions. Select the box that gives it read permission, and click apply. Do NOT give perflib write permissions. It's not necessary, nor is it wise.



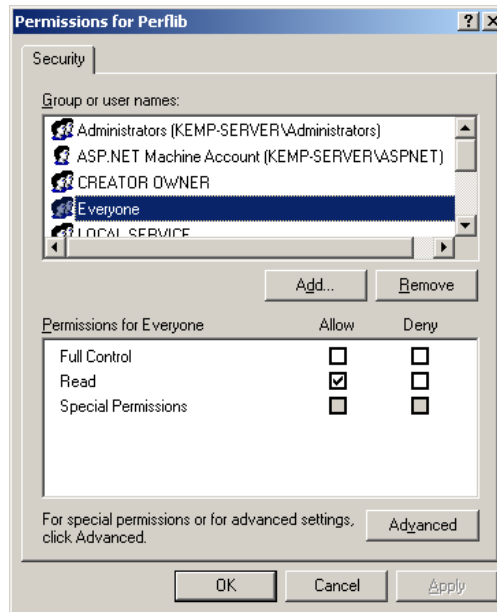


Figure 4: Perflib Permissions

At this point, you're now done. You can test out the setup by bringing up the URL in any browser. You should immediately get back a number. You can use the browser to help you tweak the results reported.

Repeat

Repeat the above steps for each real server that you wish to include in adaptive load balancing. Note that while not all real servers configured on a LoadMaster necessarily need to have an agent installed, all of the real servers in a virtual service that is configured for adaptive load balancing does need to have the agents configured.

Setting up the LoadMaster

In the LoadMaster Web User Interface, select the Rules and Checking link in the left navigation, and then the "Check Parameters" menu. In there, you'll find the configuration options for adaptive metrics.



Figure 5 Adaptive Parameters Configuration

Adaptive Interval determines how often the adaptive value is pulled. The Adaptive URL tells the LoadMaster where to find the value, and port specifies which port the web server is answering on.

If your webroot for “/” is `C:\inetpub\wwwroot\site1\`, and your agent located in `C:\inetpub\wwwroot\site1\LMagent\LMperfagent.exe`, then you would change Adaptive URL to `/LMagent/LMperfagent.exe`.

When you enter the Adaptive URL, hit enter, and the changes will be saved.

The Min. Control Variable Value is a percentage that specifies a threshold below which the balancer will switch to static weight-based scheduling, i.e. normal Weighted Round Robin. The value is a percentage of the maximum load, but maximum value permitted is 50. The default is 5.

The Min. Weight Adjustment Value is a percentage that specifies the minimum weight that can be assigned to a server, as a percentage of the initial static weight it was given. For example, if the server was given 100 to start with and this adjustment value is 10, then the weight will never be set to below 10. A value of 5 is recommended, and the maximum value is 50.



Virtual Service Configuration

Once the global settings are configured and the agent installed on the real servers, you can enable adaptive load balancing on a per-virtual service basis. In the properties for each virtual service, you'll see an option for scheduling method. Select "resource based (adaptive)" to enable adaptive load balancing.

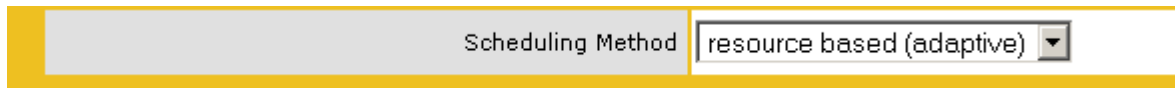
A screenshot of a configuration interface with a yellow header bar. Below the header, there is a light gray rectangular area containing the text "Scheduling Method". To the right of this text is a dropdown menu with a small downward-pointing arrow. The dropdown menu is open, showing the text "resource based (adaptive)".

Figure 6: Adaptive Load Balancing Option

