

Alpha Anywhere Standard Application Server

High Capacity Server Systems

The standard Alpha Anywhere Application Server has adequate capacity for many web systems. However, Alpha Software recommends using the IIS based Alpha Anywhere server system for larger production systems or systems requiring high capacity and the ability to scale.

Hardware Recommendations

Alpha Anywhere is a Windows based system that will run on various Windows versions. The minimum requirement is Windows 7 or Server 2008 R2. A 64 bit operating system is suggested. However, it is recommended that the latest desktop or server operating systems are used for the best performance.

Alpha Anywhere is a 32 bit program that can normally use up to 4 GB RAM and runs on a single processor core for each instance. The program also supports a dedicated print server that is only used if the application generates PDF based documents. The print server must run on a separate processor core and typically will run well with 2 GB RAM.

It is suggested for maximum performance that the machine running the Application Server has at least 1 core and 4 GB RAM per Application Server instance running. If the web application generates PDF based reports, then the recommendation is 4-6 GB and 2 cores for each instance. Additional RAM and processor cores should be allotted for the operating system. Some processes used by the standard server can use additional resources, so additional processor cores and RAM may improve performance in some configurations.

The Application Server will run adequately on fewer resources, but may see degraded performance under heavy loads. It is recommended that other programs running on the same machine should be limited to required programs related to the web application.

A single machine can often run multiple instances of the Application Server. See information below on running multiple instances of the applications server on a single machine.

Application Server Installation

The Application Server installation requires the install file and a license for each machine on which the program is installed. The install must be done by a user with full administrative privileges to make sure all support programs are installed, and all controls are registered. In some rare instances with restricted user privileges, the install may fail or some support program may fail to install. In that instance, UAC can be turned off to install the program and then turned back on after install.

The Application Server should normally be installed in the default file location unless there is a compelling reason to install elsewhere. It should normally be installed with all default settings.

If the application being used connects to an external data source such as MySQL or MS SQL Server, the Application Server will use ODBC to make the connection. Therefore, any required drivers for the data source must also be installed.

Configuring the Application Server

In most installations, all of the default settings are appropriate and no changes are needed or desired.

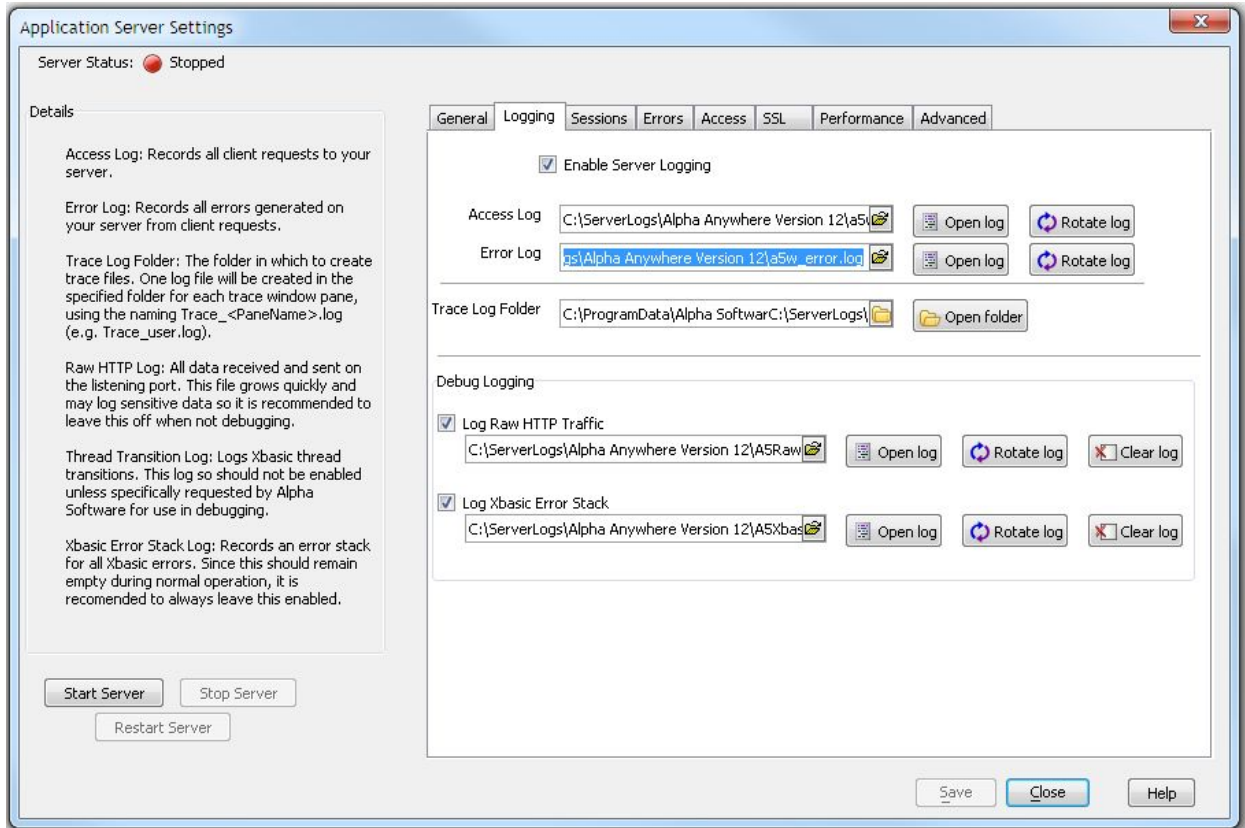
Some changes are suggested if there are multiple instances of the Application Server running on the same machine.

There is information in the help on [running multiple instances of the Application Server](#). The link includes information on setting up multiple IP addresses and also using dedicated configuration fields for each instance.

The default location for server log files is initially set in a folder created during the installation process such as:

```
C:\ProgramData\Alpha Software\Alpha Anywhere Application Server Version  
12\Installations\CProgramFilesx86a5V12ApplicationServer.
```

This folder also contains the default configuration file (ApplicationServerConfig.xml) for the Application Server. It is useful to create specific folders for each instance if running multiple instances to separate the data for each instance. These folders can identify the specific server to segregate any files created.



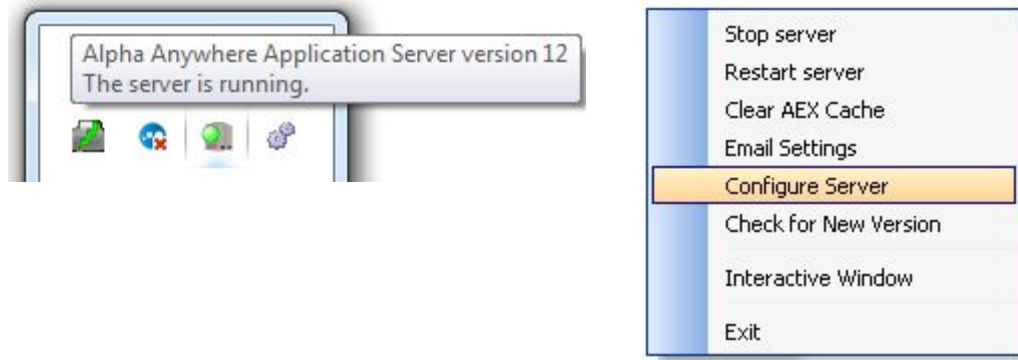
In the example above, the log files use the default file names with a path specific to this instance.

Additional information can be found in the help at [Configuring and Running the Server](#). Some screens and settings in the current version of Alpha Anywhere may be different that shown in the documentation.

Running the Application Server

The standard Application Server does not run natively as a service but is designed to allow being run as a service. The link above on running multiple server instances includes information on configuring a service to run the Alpha Anywhere Standard Application Server. Additional information can be found at [How to run as a Service](#). Programs such as AlwaysUp can also be used to run as a service. The service should be run with a valid administrative user login and not run as a local system account as a local system account will not normally have printing rights.

No user interfaces to configure the standard Application Server are normally available when running as a service. An instance can be started manually to open a configuration screen to edit the settings. A small ball icon will show in the Windows Taskbar when the Application Server is started manually. The icon has a right click menu to access the configuration and other options.



Application Server Load Capacity

Alpha Software does not publish any performance standards for the Alpha Anywhere Standard Application Server as the actual load capacity is dependent on the environment and the application design. The actual load capacity is limited by the number of requests in any time period, the time the server needs to respond to the requests, and in some cases, the CPU load available.

Factors that impact this capacity include the mix of requests (number of images for example vs the number of pages), the response time of a data server, the design of the components and web pages themselves (complex pages take longer to generate), and the number of people using the system. Access loads tend to vary greatly during any given day.

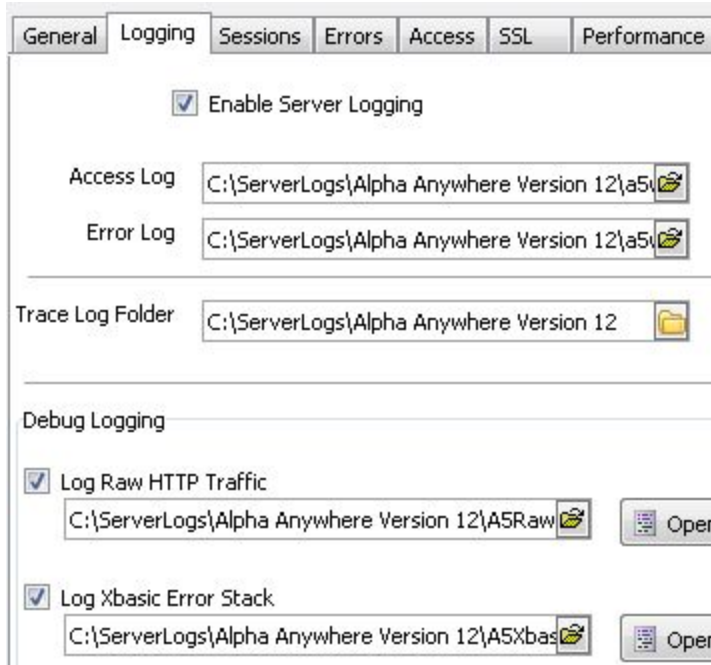
Typical systems have shown through testing that a single standard Application Server can handle 125-150 requests per minute assuming reasonable response times and a mix of requests. This typically can equate to 25-50 people using the system depending on the level of activity for each user. These numbers can vary greatly between systems and should not be considered as accurate measures. Only testing in a live system can determine the actual capacity of a system.

Server performance can also be impacted by network issues and other programs running on the same machine that may limit data transfer times or limit resources available to the standard Application Server to maintain its performance.

Testing the Application Server

The Application Server has the ability to manage overload conditions under peak loads. This will normally be presented as a slowdown in response time. The normal methods to evaluate server performance include turning on server logs to record activity and monitoring memory usage and CPU activity in the Windows Task Manager and similar monitoring tools.

There are four basic server logs that can be used.

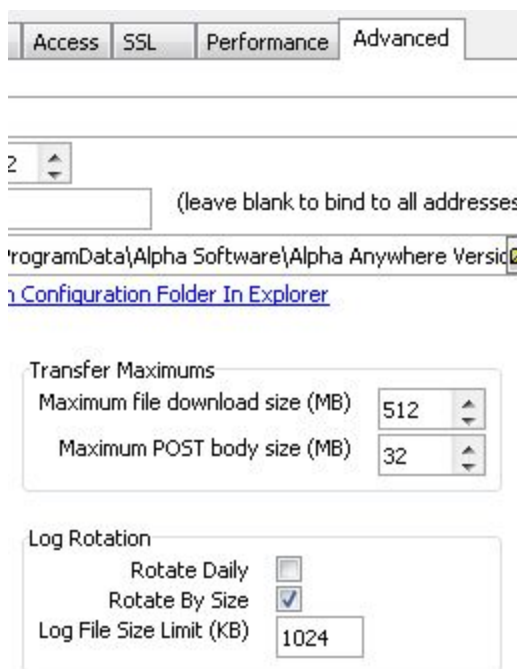


The “Log Xbasic Error Stack” Log should always be turned on. It should normally be empty on a well behaved system, but will record any unexpected code errors.

The option to “Enable Server Logging” will turn on basic access and error logs. These are typically used to determine the basic loads on the system and any errors generated. They usually are only turned on for testing and if a performance problem is identified.

The “Log Raw HTTP Traffic” should only be activated if requested by Alpha Software for analysis purposes. It includes all requests and the responses sent.

The Alpha Anywhere program includes some built-in methods to analyze access logs and graph loads using a function named “a5w_analyzeAccessLogs()”. Information on its use and expected results can be provided on request. Alpha Software can also evaluate the raw log and other logs to identify performance issues.



If logs are turned on, they should be set to rotate on size or daily to keep file sizes down.

It is also possible to determine some performance factors using the Windows Task Manager and similar tools. The Process and Performance tabs can give information on the CPU percentage in use on the core used by a standard Application Server instance, as well as the Memory used by the process.

It is normal to see 100% CPU on the core during startup for a short period, and also momentary peaks at or near 100% at peak loads. The percentage should drop to 0% when the server is idle.

Normal peak memory usage should be in the range of 400,000 K or less, although some systems can sustain higher values. The Private Working Set should be much

lower, although it may peak at high load periods. The Private Working set should drop some amount after some idle time.

Multiple Application Servers and Load Balancing

Load requirements may require using multiple Application Servers. There are multiple configurations possible depending on the hardware available and the preferences of the people managing the system. As mentioned previously, multiple instances of the standard Application Server can be run on a single machine. This is typically limited by the resources available on the machine. However, it is advised to run some instances on a separate machine if possible so one server can be rebooted without disabling the whole system.

Multiple instances will require some type of load balancing equipment. The main requirement is the system should be configured to maintain persistence or session affinity. This means a user will use the same server instance for the full life of the user session. This can be configured multiple ways depending on the local balancing appliance, but is typically done by IP address or a tracking cookie.

If the system requires using SSL, the SSL management should be offloaded to the load balancing appliance if possible.

The Alpha Anywhere Application Server for IIS server should be used for high capacity systems that may need to scale.