TwonkyServer™
UI Configuration
User Guide

Copyright and confidentiality information

TABLE OF CONTENTS

TwonkyServer UI Configuration User Guide

Overview

TwonkyServer™ is a software application that scans your media and provides it to other connected devices on your network using a protocol called UPnP A/V (Universal Plug & Play). Other devices supporting this protocol are able to "discover" the TwonkyServer on your network and navigate through the media library, and also stream the content for playback.

Even though TwonkyServer is built on open standards, such as UPnP A/V and DLNA, it is able to support a very wide variety of client devices for playback, including some devices which don't fully support these standards. TwonkyServer does this by means of a feature called client adaptation. This allows the server to adapt its response based on the type of client device it is serving content to. This allows TwonkyServer to support a wide variety of devices including Xbox360, PS3, in addition to a wide assortment of UPnP/DLNA TVs, digital media adapters, internet radios, photo frames, etc.

TwonkyServer can support a wide assortment of content types for streaming to your devices, including:

- Audio Formats:
  LPCM, MP3, WMA, M4A, M4B, MP4, 3GP, WAV, OGG, FLAC, AAC, MP2, AC3, MPA, MP1, AIF, ASF

- Image Formats:
  JPEG, PNG, TIF, TIFF, BMP, GIF

- Video Formats:
  WMV, MPEG2, MPEG4, MP1, MPG, SPTS, MP4, AVI, VOB, DivX, 3GP, VDR, MPE, DVR-MS, Xvid, M1V, M4V, MOV, MPV, MKV, FLV, MTS

- Playlist Formats:
  M3U, M3U8, PLS, WPL

Another key feature of TwonkyServer is the ability to "aggregate" content from other servers in your network. This provides a way for the server to find other servers in your network (e.g. other TwonkyServers or Windows7 servers) and collect all of the content into a unified view making it easier for the user to see all of the home content from a single location. The aggregation feature is described in more detail later in this document.

The purpose of this document is to provide the user with a basic guideline for configuring the TwonkyServer settings using your web browser.

TwonkyServer UI Configuration User Guide

TwonkyServer provides a web configuration UI as a way to change some of the server settings during runtime. This web configuration UI is accessible from any web browser. This UI is provided to allow end-users to make specific changes to the server settings. You can find the configuration web UI at the following URL: http://ip-address-of-your-server:9000/config

When the web configuration page is launched you will see a number of links on the left side of the page (see Figure 1). These links provide a variety of configuration settings:
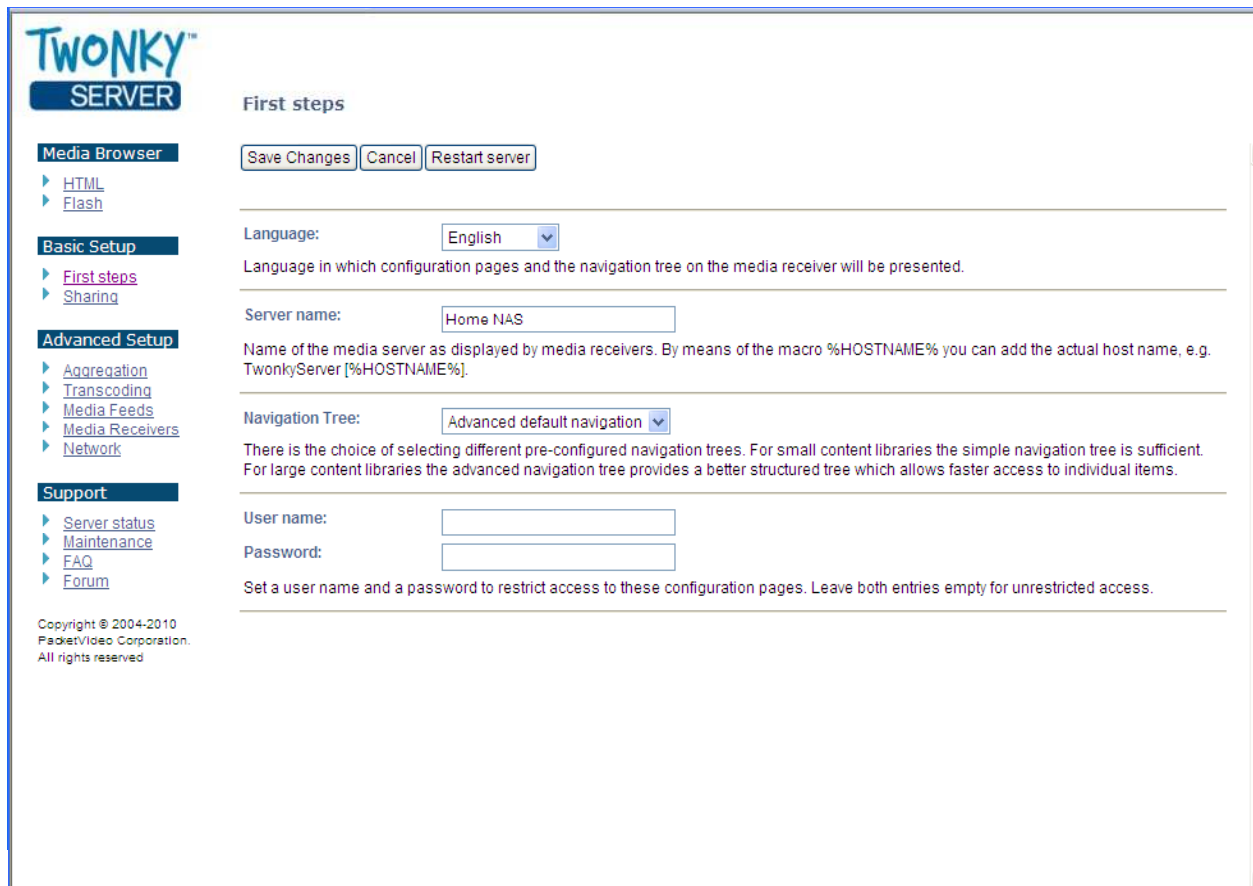
First Steps:

This link provides a page to change the **Language** settings of the configuration pages and navigation tree exposed by the server.

The **Server name** setting provides a place to specify your friendly name for the server. This is the name that is exposed to other UPnP A/V clients on the network when the server is discovered.

The **User name** and **Password** settings provide a way for the user to restrict access to the configuration pages. Leave these blank for unrestricted access to the web configuration pages.

Note: Make sure to select the "Save Changes" button after you make any changes. In some cases it may also be necessary to "Restart server" to see the changes take effect.
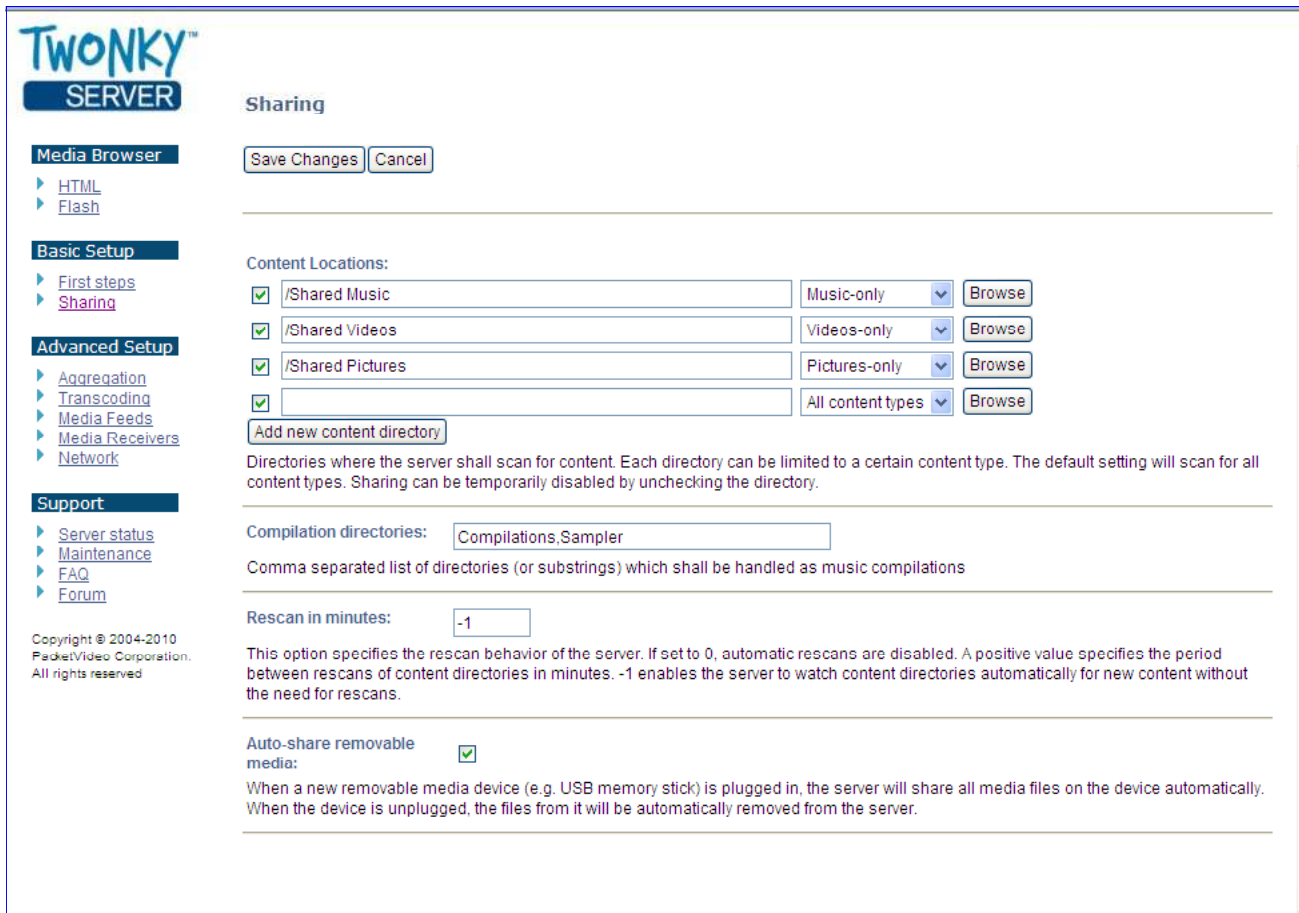
Figure 1 - Main Configuration Screen:

TwonkyServer UI Configuration User Guide

The Sharing page provides a list of the **Content Locations** (see Figure 2). These are the folders where media is scanned by the server. Each folder can be designated by media type as Music-only, Pictures-only, Video-only, or All content types. It is highly recommended to keep your content separated into different folders by media type, in order to prevent the server from listing album art JPEGs into the Photo library, for example.

The **Compilation directories** setting provides a comma-separated list of directory names (or substrings) that help the server to recognize music compilations where an album of music may have numerous artists under a single album title. Normally, users don't need to change these default settings, but they can add more entries to this list if desired.

Figure 2 - Sharing Configuration Screen

The **Rescan in minutes** setting provides control to the way in which the TwonkyServer scans for new content in the list of Content Locations folders. By setting this value to -1 it directs the server to automatically scan any new content that is placed into the Content Locations folders. This is the preferred setting since the server only rescans when it detects new content. This provides the best overall performance of the server. This mechanism relies on existing "inotify" support in the Linux kernel (if Twonky is running on a Linux platform). For Windows platforms it utilizes a different OS capability to provide the same kind of notification when new content is added.

TwonkyServer UI Configuration User Guide

Note: If the Linux kernel does not have support for inotify, then it is possible to change this **Rescan in minutes** setting to some other positive value, corresponding to the number of minutes in which the server will wait between rescans of the Content Locations folders.

The **Auto-share removable media setting** is used to allow the server to automatically scan removable media (e.g. USB memory sticks, etc) and expose their content for UPnP clients to navigate and stream. When the device is unplugged it automatically removes it from the server navigation.

Note: This feature relies on special integration with the device's USB eventing mechanism and the server's removable media API. For Windows platforms this is pre-configured.

Aggregation:

One of the key new features added with TwonkyServer 6.0 is Aggregation (see Figure 3). Aggregation allows the TwonkyServer to collect (or aggregate) content libraries from other servers in the network and provide a single unified-view of all the available media. In this way a user only needs to navigate to the main TwonkyServer to see all of the available content in the home network. The user doesn't need to bother with navigating through all of the servers to find the specific item's they're looking for. As the aggregator finds media on the various servers within the network, it is able to detect duplicate content (e.g. the same music tracks on different servers) and only expose a single copy of this duplicated content.
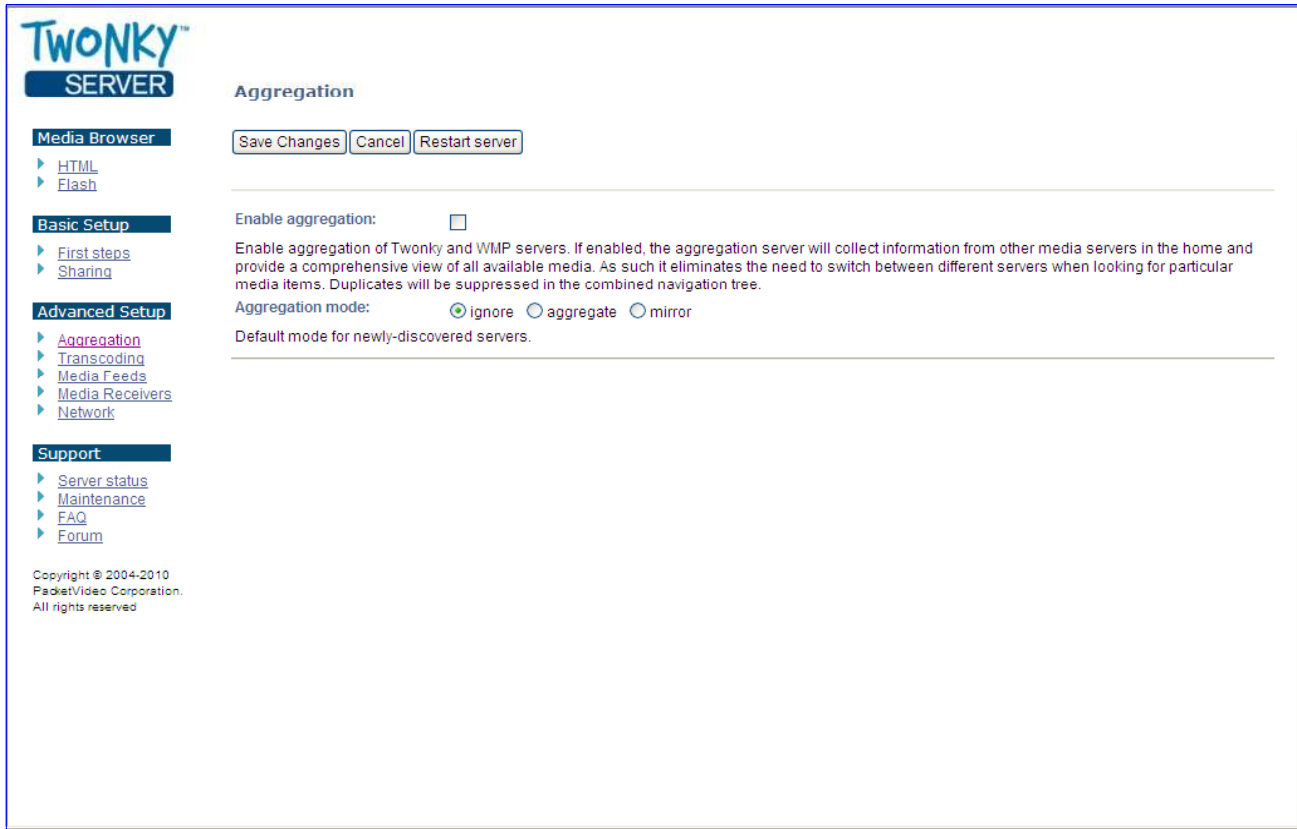
The aggregator is able to aggregate other TwonkyServers (even the ones running on Android phones), as well as Windows7 media servers (enabled when media sharing is activated in Windows7).

The Aggregation feature operates in few different modes:

- **Aggregation** - In this mode the TwonkyServer aggregates only the content metadata (artist, album name, track name, etc.) but the media content is never moved from its original location. The content files stay in their original location, but the aggregation server is able to "point back" to them for streaming, etc. If the content is being hosted from a server that is later removed from the network (e.g. content that is served from an Android phone) then the Aggregator is able to recognize when a server has left the network, and it remove the list of content from that server, from the aggregated view.

- **Mirroring** - In this mode the TwonkyServer aggregates the content metadata, and also copies the content itself over to the aggregation server. In this way, the aggregation server provides a way to automatically consolidate the aggregated content onto the main TwonkyServer. This provides a useful feature for making a backup copy of content from TwonkyServers that may be running from Android phones, for example.

By default, the aggregation mode is turned off. To enable aggregation, select the "Aggregation" link from the TwonkyServer web config page, then select the "Enable aggregation" button.

Figure 3 - Aggregation Configuration Screen

It is also possible to select the default mode for any newly discovered servers: ignore (no aggregation), aggregate (collecting the metadata from the other servers), or mirror (collect the metadata and also copy the content over to the main aggregation server).
As the aggregator begins to discover servers in the network they will be listed on this Aggregation page of the web configuration (see Figure 4). For each server it will display the number of content items that are being aggregated from each server.

For each server discovered by the aggregator it is possible to "ignore" individual servers. This is a useful feature as it may not be desirable to aggregate the content from all discovered servers. The aggregator is flexible enough to offer various ways to configure the feature to suit the needs of the end user.

TwonkyServer UI Configuration User Guide

Figure 4 - Aggregation Configuration Screen Options

Transcoding:

The TwonkyServer provides an API for integration of transcoders for the server to utilize. Transcoders allow the TwonkyServer to convert source content into other formats on behalf of client devices, which may support a limited number of formats for playback. Common uses of transcoding would be for content such as video, where the source video is converted into some other format that is needed by the video playback device. The TwonkyServer provides a JPEG transcoder, which is pre-integrated with the server. This JPEG transcoder allows the TwonkyServer to scale JPEG images to fit the needs of the client device, since some clients will only support specific JPEG resolutions. This JPEG scaler is also used to provide thumbnail preview images.

The transcoding feature can operate in either "on-the-fly" mode, where the content is transcoded as needed, or background mode, where the content is converted as a background process and stored on the server for later use.
The Transcoding page on the web config UI allows the user to enable background transcoding for any transcoders that have been integrated with the TwonkyServer (see Figure 5). The transcoding options are "grayed out" for other formats where the transcoder is not pre-integrated and not selectable by the end user.
The TwonkyServer does not provide transcoding software packages pre-integrated, with the exception of JPEG scaler. However, OEMs are free to integrate transcoding packages as they wish, so long as the platform has adequate CPU performance and RAM to support the type of transcoding desired. Most embedded platforms can support JPEG scaling in real time with little trouble, but real-time-video transcoding requires much more CPU performance.

TwonkyServer UI Configuration User Guide

Figure 5 - Transcoding Configuration Screen

Media Receivers:

The Media Receivers page provides a list of all of the UPnP media receivers discovered by the TwonkyServer. One of the unique features of the TwonkyServer is "client adaptation". This feature provides a way for the TwonkyServer to make adaptations to the interactions with client devices based on the type of device it is interacting with. The TwonkyServer uses this client adaptation layer to make adjustments to things such as photo scaling, handling of specific mime types, changes to header information, and a variety of other things that allow the client device to work with the best possible user experience when interacting with the TwonkyServer. In some cases this can result in the device "working well" versus "not working at all" with a more generic UPnP media server that is not able to make adaptations like this.

Client adaptation is made possible by a small database contained within the TwonkyServer that is maintained with various adaptations based on the client device. When the TwonkyServer discovers various clients on the network it utilizes this database to make these adjustments as needed.
PacketVideo continually provides new additions to this client database as more and more devices are tested and characterized in our labs. As new updates to the TwonkyServer are released, this database is also updated with the most current version based on our own testing. This is one of the reasons why the TwonkyServer is able to support such a wide variety of devices.

The Media Receivers page (see Figure 6) maintains a list of discovered client devices on your network by MAC address. The current IP address is also listed along with the client adaptation setting (showing the client type), and the navigation setting for each client device.

TwonkyServer UI Configuration User Guide

The user can change the navigation setting to one of three options (Advanced default navigation, iPod like, or by folder). These three options affect the style in which the TwonkyServer exposed the content metadata to each client device.

The Client Adaptation setting is also user-selectable, although it is not recommended to override the automatic value since it may result in unexpected behavior. It is usually best to allow the server to automatically select the best client adaptation setting. For some devices this setting will default to the setting "Generic Media Receiver" if there is no specific adaptation recognized in the client database. In some cases the server will use this setting initially until the device is recognized, and provided with a specific client setting from the database.

Figure 6 - Media Receivers Configuration Screen

TwonkyServer UI Configuration User Guide

Network:

The Network page (see Figure 7) provides a button called "**Restart on NIC changes**". This button allows the TwonkyServer to restart itself if there are any changes to the Network Interface Controller. The Network Interface Controller (or Network Interface Card) is the piece of hardware that connects your device to the network. (LAN connection).   In order to work properly, the TwonkyServer needs to know the current network connection details for the device it is hosted on, which is made available by the NIC. If this information changes for any reason (e.g. the IP address changes due to a router reboot, or more than one NIC connection becomes active) then the TwonkyServer needs to restart in order to use the current NIC settings.

It is highly recommended to leave the "Restart on NIC changes" activated at all times.

Note: The TwonkyServer is able to bind to more than one Network Interface simultaneously. When the TwonkyServer starts it looks for all of the active network interfaces and binds to them. The Server status page provides a list of all of the network interfaces that the server is bound to. By default it always binds to 127.0.0.1 which is the "loop back" IP address.

Figure 7 - Network Configuration Screen

TwonkyServer UI Configuration User Guide

Server Status:

The Server status page provides a summary of several important items pertaining to the TwonkyServer's current state.   Here is a short description of each item on this status page:

- *Version:* This is the specific software version of the TwonkyServer software that is running.
- *Cache memory:* This is a measure of the size of data being stored in the Cache folder of the server. The Cache folder is used to hold any transcoded content, such as scaled JPEG images (created by the JPEG scaler, thumbnails, or other types of transcoded content created by the server on behalf of the client devices that are being served.
- *Last database change:* This provides the timestamp of the latest re-build of the TwonkyServer's internal database. The TwonkyServer maintains this database with information about the media content that it scans on the system. This provides all of the content metadata (title, album artist, genre, etc) in addition to the location of the content for streaming.
- *Music Tracks:* This lists the total number of music tracks found by the TwonkyServer from the shared folders on the system.
- *Photos:* This lists the total number of photos found by the TwonkyServer from the shared folders on the system.
- *Videos:* This lists the total number of videos found by the TwonkyServer from the shared folders on the system.
- *License information:* This indicates if the TwonkyServer is licensed on your system. For most embedded systems this will simply read "The Server is registered". For other systems (such as a PC build) this will show the license key being used.
- *Server uptime:* This shows the length of time that server has been running since the last restart.
- *Server build date:* This indicates the build date for the TwonkyServer that is installed on this system.
- *Server restart pending:* This indicates if the TwonkyServer needs to restart due to some changes that have been made on the web configuration pages. The TwonkyServer needs to restart in order for certain types of changes to take effect.
- *Network Interfaces:* This indicates all of the current network interfaces that the TwonkyServer has bound to. By default the TwonkyServer binds to the loopback IP address 127.0.0.1, however, there are usually one or more other IP addresses listed here, indicating the IP address for the host machine that the TwonkyServer is running on.

Maintenance:

The Maintenance page (see Figure 8) provides several useful TwonkyServer commands for quick access:
- *Save Changes:* This command is used to save any changes to the settings that have been made on any of the other pages of the web configuration UI.
- **Restart server:** For some changes, it is also necessary to restart the server (in addition to the Save Changes) for the desired changes to take effect.
- *Rescan content directories:* This command is used to direct the TwonkyServer to rescan all of the shared folders (see Content Locations from the Sharing page). Rescanning is normally not necessary if the server is automatically detecting any new content (by setting the rescan in minutes setting to -1 on the Sharing page). However, there may be occasions when it is desired to direct the TwonkyServer to rescan immediately.

- **Rebuild database:** This command provides a way to direct the server to completely rebuild its internal database. Keep in mind that using this command will remove any metadata that may have been added to the server from TwonkyManager. For example playlists, slideshows, and user ratings (added via TwonkyManager). During the rebuild process, the TwonkyServer will delete its old database and rescan all of the media content to build the new one. This process can take a significant amount of time for a large library.
- **Reset to defaults:** This command will reset all of the TwonkyServer settings to the default values. These default values are found in the twonkymedia-server-default.ini file, which is located in the installation folder for the TwonkyServer.

The **Logging** button provides a way to enable logging for the server. This may be useful for troubleshooting. The "Clear logs" and View log file buttons are provided to allow the user to clear or view the log files from their browser.

Figure 8 - Maintenance Configuration Screen

TwonkyServer UI Configuration User Guide