Defining And Managing User Accounts

System administrators run FTP servers to allow numerous other people to send and receive files to and from a single server. Each person who will use the server may be treated differently, be given different views of the server, or have different access rights. In addition, as an FTP server administrator, you may wish to restrict unauthorized people from accessing the server at all, or at least offer unknown users limited server access.

The process of requiring that new users supply a username and password to gain access to a file server is called "authentication". To manage this process, Rumpus allows you to create and manage "user accounts", records that are associated with a username and include a variety of options that will serve to define the capabilities each person who accesses the server is given.

Put simply, Rumpus allows you to define user accounts which will be used to enforce access to your server based on a supplied username and password, and which will allow you to give different levels of access to different people. Think of user accounts as "preferences" that can be set individually for each person who will be logging on to your server.

Anonymous Users

Allowing public users to access an FTP server is called granting "anonymous" access. A special user account is defined in Rumpus, called "ANONYMOUS", which allows you to define the access given to unknown public users. In most cases, anonymous access should be restricted to logging in, retrieving directory listings and downloading files, at most. Permitting anonymous users to upload files, for example, is usually considered dangerous because without additional access restrictions, anyone on the Internet can store files on your server.

Standard practice is to supply an e-mail address as the password when a user logs in anonymously, though this isn't necessarily required. If you would like to require that the client send an e-mail address as the anonymous password, enter an "at" sign ("@") as the anonymous user account password.

The anonymous user account can't be deleted, but in all other ways the ANONYMOUS user account functions exactly as any other account you will define.

Defining User Accounts

Defining and managing user accounts is done using the "Define Users" window. In the "User Accounts" list at left you will see each account name, and at right is the detailed settings for each account. Select a user account from the list, and the detail information will change to reflect that account's settings.

To create a new user account, click the "Add User" button just below the User Accounts list. A sheet will drop down, allowing you to supply the account name and password. If you select an existing user account before clicking the Add User button you will have the option of using the permissions and settings of the selected account as the starting point for defining the new account. For example, to create a new account for "bob" that will be similar to the user account "tom", select the "tom" account in the User Accounts list, then click the Add User button. On the New User sheet, check the "Start With Settings Of Selected User Account" option, and the account "bob" will inherit the settings from the account "tom".

The sheet also allows you to specify the user account password. Be sure to enter the password carefully, as the password supplied by connecting users will need to match exactly. If you would like to allow users to be able to login with any password, set the account password to an asterisk "*". Or, to require a password that matches a standard-looking e-mail address, set the password to "@". Once the user account has been created, the password pop-up menu will display whether the account is set to allow any password, an e-mail address as password, or an exact match password. The pop-up menu can also be used to select among these options as needed.

User Folder

When a user logs in to the FTP server, they will immediately be "placed" into their User Folder. This is the folder on your server's hard drive that will serve as their top-level home folder, and is also sometimes called a "drop folder". This allows you, for example, to have multiple users, each with an account on your server, who are unable to access or even see the contents of other user's folders.

The User Folder can be set to any folder on your hard drive by entering the full path to the folder or by clicking the folder button just to the right of the User Folder field. However, we strongly recommend that you create a user folder inside of your FTP Server Root folder, and assign that folder to the account. This will not only simplify server management, but also improve security by localizing all FTP User Folders within a single location on your hard drive.

If the user should be given access to the FTP Server Root folder, rather than a unique drop folder for their account, then click the Home button to set the User Folder to "ROOT".

Management Options

Privileges

Each of the checkboxes in the Privileges grouping defines the actions that the user can and cannot perform. For example, you can temporarily disable an

account by un-checking the Permit Login option. The account will remain in the Rumpus database and can be re-enabled at any time, but when this checkbox is unchecked, the account will be effectively disabled.

Other permissions include the ability to download, upload, delete, and see files and/or folders.

The privileges defined for the user account may be over-ridden on a folder-by-folder basis. To define different restrictions for one or more specific folders, create a "Folder Set" and then select it from the "Apply Folder Set" pop-up menu.

Trigger Upload Notice

If you would like to have a notice triggered when the selected user uploads a new file, create an Upload Notice and select it from this pop-up menu.

Trigger Download Notice

If you would like to have a notice triggered when the selected user downloads a new file, create an Upload Notice and select it from this pop-up menu.

Access Restrictions

Maximum Folder Size

A maximum size can be set on the drop folder, which restricts the amount of hard drive space available to the user. The Maximum Folder Size is set in megabytes, and once the maximum size has been reached, the user will be unable to upload additional files until old files are deleted. Note that enabling this option forces Rumpus to compute the user's user folder size before each new file upload. For large folders, this can take some time and slow down file upload processing.

Maximum Concurrent Connections

The Maximum Concurrent Connections setting will limit the number of simultaneous connections the user is allowed to make to the server. Enter the maximum number of concurrent connections you wish in the text box. This limit is enforced only when the checkbox is checked. One "connection" is counted as one data/control connection pair.

This setting prevents FTP client software from making an unlimited number of connections to the server. Some client applications will open a separate connection for each file requested, resulting in an unreasonable number of connections from a single client machine. Limiting this prevents a single user from using up more than a fair share of the maximum number of connections to the server.

Maximum Upload Rate

This setting allows limiting the maximum transfer rate of file uploads for each user account. The limit is applied on a per-connection basis, so users who make multiple connections will be able to transfer data at the maximum rate for each connection. Enter a maximum transfer rate in kilobytes per second into the text box. The limit will be enforced when the checkbox to the left is checked.

This setting prevents a single user with a high-bandwidth Internet connection from using most or all of the server's connection bandwidth.

Note that the transfer rate limit is not exactly enforced. Actual transfer rates may be +/-15% of the setting. This is due to buffering done by the TCP/IP networking stack and devices in the transport chain. Inaccuracy is also due in part to inaccuracies in counting the transfer rate in FTP client applications.

Maximum Download Rate

This setting limits the maximum speed of file downloads for each connection made by the specified user. The option works the same way, and is subject to the same limitations, as the Maximum Upload Rate, described above.

Maximum Upload:Download Ratio

A common feature of bulletin board systems, the upload:download ratio allows the administrator to require users to upload a certain amount in a ratio with the amount that they download. In order for users with an upload:download ratio enabled to download files, they must first upload a file to the server. Thereafter, downloads are only permitted when the amount of data uploaded exceeds the amount downloaded by the specified ratio.

Enter the ratio of uploaded to downloaded bytes in the text box. The ratio will be enforced when the checkbox to the left is checked. If the ratio is exceeded during a download, Rumpus will allow that file to be downloaded completely. It will not stop a download resulting in a broken file. If a subsequent download is attempted, Rumpus will return an error message to the client indicating that the ratio has been exceeded.

Note that the operation of the ratio is based on the number of bytes uploaded, not by the number of files uploaded. For example, if a user 's upload:download ratio is set to 5, and they upload one file of 37823 bytes, they will be allowed to download any number of files totaling not more than 189115 bytes, or approximately 184kB.

Account Type

User accounts can be created so that they will expire at some future date. This allows you to create temporary accounts without having to remember to delete or deactivate them later. By default, user accounts are "Permanent" and will never expire, but you may also choose "Disable" to have the "Permit Login" privilege turned off on a given date or "Delete" to have the user account removed entirely. If you select Disable or Delete, specify the date on which the user account should be disabled or removed in MM/DD/YY format.

Security Settings

Allow User To Move Out Of User Folder

In most cases, when you assign a User Folder for an account, the user should be given server access only to that folder, or folders within that folder's hierarchy. However, if you wish, the user may be "dropped" into their user folder only as a starting point, after which they may move up and out of their user folder and into the FTP Server Root folder. Note that for this option to work as expected the User Folder must be located in the FTP Server Root folder. If it is not, then "moving up" from the User Folder into the Server Root is impossible.

Save Files With Custom Ownership/Permissions

Files uploaded via Rumpus will be saved with the owner set to the same user that was used to launched Rumpus. Since Rumpus is run as "root", the file owner for all uploaded files will also be "root". In addition, a default set of access permissions are applied to files which may restrict access to them when you copy or open them directly on the server.

Rumpus allows you to override the system default privileges, giving you full control over the restrictions placed on working with uploaded files. When enabled, the "Save Files With Custom Permissions" option will save files with "read", "read & write", or "none" privileges for the file owner, group and "everyone", as specified.

In addition, you have the option of specifying the Unix file system owner of files uploaded to the server or newly created folders. Simply specify the name of any user account on the Unix system (not a Rumpus user account) and uploaded files will be assigned to that user. If you prefer, check the “Set Owner To Parent Folder Owner”, and Rumpus will automatically set the owner of new files to the owner of the folder in which they are saved. In the event that Rumpus is unable to determine the owner of the parent folder, the specified “Owner Name” will be applied instead.

Important! The owner specified in the “Owner Name” field must match a user account on the Mac OS X system. In Unix terms, Rumpus issues a “chown”

(CHange OWNer) command on any file uploaded to the server or new folder that is created. If the owner name does not exactly match an existing user on the system, the file will retain the owner of the user account used to launch Rumpus (usually “root”).

Usage History

The History tab displays statistics about past FTP access by the user. The statistics tracked are self-explanatory, but please note that the statistics are updated only when a user logs out of Rumpus. In other words, the display does not reflect activity that has taken place in any active FTP session. (For this information, see the main Rumpus Status window.) When a user logs out of Rumpus, however, the display is immediately updated.

The account history can be reset by clicking the "Reset History" button. It is important to note, however, that historical information is not used exclusively for display purposes. For example, the "Bytes Uploaded" and "Bytes Downloaded" values are used to determine whether or not a user can download a file when an upload/download ratio is being enforced.