Server Security

How to protect your server from unauthorized use.

Is Rumpus Secure?

This is a common question, but unfortunately there isn't a simple answer. Rumpus can certainly be used as part of a strategy in maintaining a well secured server, but it is crucial to understand that the security of your server depends not on the software, but on how it is used. Like a hammer, for example, Rumpus is a tool that can be used safely, but can cause damage when common sense precautions aren't taken.

It is also important to understand that there is no such thing as an unconditionally secure server. Stating unequivically that a server is secure is simply impossible. Instead, server administrators need to take reasonable security precautions to achieve a level of security necessitated by the content of the server.

In other words, first consider the importance of the data on your server and the value it might have to an unauthorized user or the cost you would incur if the data were destroyed. The effort you spend securing your server should reflect the nature of the data you are protecting. This is no different than in the physical world, where greater security measures are employed by banks than by grocery stores, for example. Unfortunately, this common sense concept is often overlooked due to sensationalized stories of Internet theft and a general lack of understanding about how the Internet works.

Practical Security Measures

The key to avoiding problems is to use common sense and remain diligent as you administer the server. Presented below are a few tips that will help you maintain reasonable security. Of course, this isn't an exhaustive list of every security issue you should consider. Again, common sense and an understanding of your actual security needs should guide your overall strategy to keeping your server safe.

Use Care When Creating User Accounts

One big advantage of Rumpus over some other FTP servers is that you don't have to create system user accounts in order to grant new users access to the Rumpus server. Right away, this prevents users from using their FTP login account to access the server via SSH, AppleShare, or other protocols that may be enabled on your server. However, be sure to set new user account Home Folders to well contained areas of the hard drive, and limit FTP privileges to those capabilities required by the user. Don't use intuitive passwords that are easy to guess, and remove old accounts when they are no longer needed. When creating new accounts, use common sense to assign access rights based on the trustworthiness of the person that will be using the account.

Watch Out For Aliases

Mac OS aliases are extremely handy for giving users access to areas of the hard drive outside of their Home Folder as needed, but they need to be used carefully. One misplaced alias can change a user's access from a tightly controlled home folder to free reign over your entire network. And remember that more than one alias can be used in a path. For example, if the folder "Bob" includes an alias to the top level of the server hard drive, and an alias to Bob's folder is put in the Home Folder "Tom", then Tom will instantly be granted access not only to Bob's folder, but the entire server hard drive.

Deploy A Firewall

Mac OS X includes a firewall that is simple and effective, and enabling it is perhaps the most basic security precaution your can take. When first getting started with Rumpus, it often makes sense to disable the OS X Firewall to keep it from interfering with your server as you bring it on-line. However, once your server is running and tested, turn the firewall on. You'll need to make holes in the firewall, of course, to allow FTP and WFM traffic, usually on ports 21, 80 (or 8000) and 3000-3008. To enable and configure the firewall, open the System Preferences window, select the "Sharing" panel and flip to the "Firewalls" tab. If you have trouble, contact Maxum Technical Support at "support@maxum.com".

Minimize Running Applications And Processes

Many server attacks make use not of a single insecure server process, but several processes used together in unexpected ways. Reduce this risk by minimizing the number of programs running on your server. And don't think that Rumpus is the only process running... Open the "Activity Monitor" ("/Applications/Utilities/Activity Monitor") and you'll find dozens of active programs running on your server.

Manage Physical Access To The Server

Physically securing a server is often overlooked, but all of the firewalls and software controls in the world won't stop someone from messing with your system if someone can sit down at the keyboard and open the Finder. If necessary, keep the server in a locked closet or server room. Also, turn off Mac OS X Automatic Login (See "Login Options" on the "Accounts" System Preferences window) and get in the habit of leaving the server at the system login window. The computer doesn't need to have a user logged in for Rumpus to run, and this way, even if someone does gain physical access to the computer, they won't be able to get into the Finder without a system username and password.

FTP and HTTP: Cleartext Protocols

FTP (and non-SSL encrypted HTTP) is generally considered insecure because the commands and data are transferred in clear text between the client and the server. This means that any network device between the client and the server can read, capture and even alter the data being sent. While this is potentially a serious problem, keep in mind that not every computer on the Internet can "listen in" on every data transfer. A reasonable analogy is the global telephone system, where phones can be "tapped" by patching into the network at some point between the two ends of the connection. And just as the easiest way to listen in on a phone conversation is to pick up an extension phone in the same house, note that other computers on the same local network can usually be used to intercept communications fairly easily.

Making the fact that basic FTP and Web data streams are unencrypted even worse is that the visibility of the text includes the user's name and password. This makes it possible for a hacker to intercept just the initial login portion of an FTP session in order to discover a secure account password.

Encrypting FTP and HTTP sessions is, of course, possible. Web browsers typically include well-integrated SSL capability, while this feature is less common in FTP client software. Rumpus WFM and FTP sessions can be tunneled in SSL to provide secure communication, and these options are described in the "Secure Transfers" article in this package.

User Authentication

The most obvious security feature of most Internet servers is user authentication by name and password. Rumpus fully supports authentication through the creation and management of user accounts. See the "Define Users" help page in Rumpus and the "User Accounts Primer" article for full details. Rumpus user accounts not only allow you to require users to supply a name and password, but restrict access to specific folders of the server, set additional access restrictions, track individual server use, and more.

IP Address Restrictions

In addition to requiring users to authenticate themselves, Rumpus can also restrict access to specific computers. Rumpus allows you to specify, by IP address, troublesome computers and will deny them even the attempt at logging on to your server. In addition, Rumpus can actively track potential attacks and block offending computers automatically.

A list of "blacklisted" client IP addresses is managed using the "Blocked Clients" window, shown below.
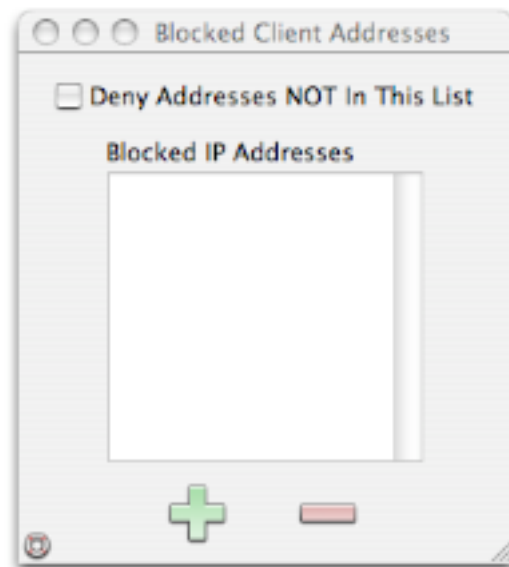
The Blocked IP Addresses Window

Any connection attempt made from a computer with an IP address in this list will be immediately terminated. Subnets can also be specified simply by entering "0" for one or more parts of the address. For example, to block access from the address "192.168.1.33" you would enter the entire address, but to block all access from any address on the entire "192.168.1.XXX" subnet, you would add "192.168.1.0" to the list.

To add an address to the "Blocked IP Addresses" list (also known as the "block list"), click the "Add" button and enter the address on the drop down sheet. To remove addresses from the list, select the address in the block list and click the "Remove" button.
For additional information on managing the block list, see the "Blocked Clients" help page in Rumpus.

Additional Security Features

Rumpus offers a number of other security options, including the automatic addition of computers guessing passwords to the "block list" and the ability to enable or disable FTP features that can be used in various types of attacks. These options are available on the "Security" tab of the FTP Settings window, and are described in detail on the "FTP Settings" help page.