HerculesAdminTool

“HAT”

User Guide

Written: 2015-08-28

Last update: 2016-01-25

Author/Maintainer: Jguy

Version: 0.6.3

Table of Contents

[About this guide 4](#_Toc461721230)

[About the HercAdminTool 5](#_Toc461721231)

[Required/Optional Dependencies 5](#_Toc461721232)

[Installation 6](#_Toc461721233)

[Databases 7](#_Toc461721234)

[Configuration 7](#_Toc461721235)

[Config.php 7](#_Toc461721236)

[Hat.php 7](#_Toc461721237)

[HAT multi-server configurations 8](#_Toc461721238)

[Login servers 8](#_Toc461721239)

[Ragnarok Servers 10](#_Toc461721240)

[Database.php 11](#_Toc461721241)

[Setting up first user 11](#_Toc461721242)

[System Performance Monitoring 11](#_Toc461721243)

[Additional Security Measures 12](#_Toc461721244)

[Modules and basic usage of the panel 13](#_Toc461721245)

[Server Selection 14](#_Toc461721246)

[Dashboard/Server Info Module 14](#_Toc461721247)

[Account Module 14](#_Toc461721248)

[Searching 14](#_Toc461721249)

[Account List 15](#_Toc461721250)

[Account Basic Info 15](#_Toc461721251)

[Account Blocks 15](#_Toc461721252)

[Notes 15](#_Toc461721253)

[Register DB 15](#_Toc461721254)

[Global Register DB 15](#_Toc461721255)

[Storage 16](#_Toc461721256)

[History 16](#_Toc461721257)

[@cmd Log 16](#_Toc461721258)

[Characters 16](#_Toc461721259)

[Admin module 16](#_Toc461721260)

[User Management 16](#_Toc461721261)

[Group Management 17](#_Toc461721262)

[Character Module 17](#_Toc461721263)

[Searching 17](#_Toc461721264)

[Character List 18](#_Toc461721265)

[Character Details 18](#_Toc461721266)

[Items 18](#_Toc461721267)

[History 18](#_Toc461721268)

[Friends 18](#_Toc461721269)

[Pet/Homun Info 18](#_Toc461721270)

[Skills 18](#_Toc461721271)

[Database Module 19](#_Toc461721272)

[Setup 19](#_Toc461721273)

[Server Setup 19](#_Toc461721274)

[Hercules Info/Maintenance 19](#_Toc461721275)

[Appendix A – Group Permissions 20](#_Toc461721276)

[Account Module Permissions 20](#_Toc461721277)

[Character Module Permissions 21](#_Toc461721278)

[Admin Panel Module Permissions 22](#_Toc461721279)

# About this guide

This guide was written to get you started with the use and maintenance of your installation of the HerculesAdminTool. Everyone's installation environment is bound to be different so this guide will attempt to cover as many scenarios as possible.

This guide will not get into indepth server configuration or installation, or teach you how to install Hercules except in the case of where it directly pertains to the tool. It is recommended you have a basic understanding of Linux, and Hercules before attempting installation.

This guide will make use of key configuration points, indicated by the below:

|  |
| --- |
| !!! Pay attention to this !!!  It's important. |

When you see this, make sure you read what it says to avoid any further issues down the road.

Anything that is to be typed into the console will appear as italic text with a grey background in a black bordered box, and will be prefixed by the $ or # symbol, or a mysql>, indicating what permissions you need to run the command, like this:

|  |
| --- |
| # apt-get update |

Additionally, information in a blue box is backend coding functionality information, such as this:

|  |
| --- |
| ?? This is regarding information about backend coding. ??  Here it will explain how something was done or why it was done that way. |

# About the HercAdminTool

The HerculesAdminTool, or “HAT” is an administration interface for your Ragnarok Online private server running the Hercules emulator. Due to some of the features used, at this moment it is only compatible with the Hercules emulator. A rAthena version at this moment is not planned. The tool was designed to be a backend panel similar to PHPMyAdmin to administrate your Hercules server, and control user's access permissions via an easy to use interface. The group configuration contains over 60 different permission settings for each group, and with 98 accessible groups, there are 98 different ways to setup GM's with panel access on your server. The accounts in the HAT are not controlled by your access permissions in-game, rather the HAT has its own user access database with its own groups. This allows you to control GM’s who may only be support (access to ticket system) or administrators/developers or QA where they would not have access to edit an account, but would have access to change an item on a character for testing purposes.

HAT allows you to simplify maintenance by allowing you to start and stop the running Hercules server via an easy to use web interface. You can view console logs from the running server, plan and execute in-game commands, in-game broadcasts, ticket system, edit files and settings on the Hercules server.

# Required/Optional Dependencies

The HAT was developed on an Ubuntu server running the latest version of Apache and PHP. You will need to install the following dependencies to use HAT:

* Apache2
* Sendmail has to be available via /usr/sbin/sendmail
* PHP5, with modules php5-mysql and php5-mcrypt

You should also have the apache module mod\_rewrite enabled, and your per directory settings enabled to AllowOverride All.

Your server running HAT must

# Installation

You will need the mcrypt module on the webserver in order for HAT to function. On most Unix systems you can enable the extension by running the following after installing mcrypt:

# php5enmod mcrypt

If this does not work, simply add `extension=mcrypt.so` to your php.ini file.

And finally, you will need to enable the apache module rewrite, which can be done by using the following command:

# a2enmod rewrite

After the above two steps, restart the apache service.

You will also need to setup a directory from which apache will serve files and ensure the directory allows you to Override all and Follow SymLinks. I would use the directory /var/www/hercadmintool and connect via an in descript port, for this guide I’ll use 23000. I also want to use a domain such as admin.your-domain.com. You can then setup an apache site like the following:

<VirtualHost \*:23000>  
 ServerAdmin [youremail@yourdomain.com](mailto:youremail@yourdomain.com)

DocumentRoot /var/www/hercadmintool

ServerName admin.your-domain.com

<Directory />

Options FollowSymLinks

AllowOverride None

</Directory>

<Directory /var/www/hercadmintool>

Options FollowSymLinks

AllowOverride All

Order allow,deny

Allow from all

</Directory>

</VirtualHost>

Make sure you also ensure Apache server is going to listen on that new port by editing the ports.conf file!

## Databases

In the /doc/sql folder you will find three database files. Depending on your setup you will need to run them against different databases.

**hat.sql** This file contains the tables for you to be able to login to HAT and use it. Without this table, HAT just won’t work. This file can live in its own database or a separate one, if you so desire. You can configure this later.

**hat\_charmap.sql** This file contains the tables for logging character and guild edits for the individual char/map server combinations, explained below. This file must be run against your Hercules database on every database that has a character/map server.

**Hat\_login.sql** This file contains the tables for logging accounts for your login server(s). This must be run against every Hercules database you have that has a login server attached.

If you only have one server you’re using HAT for, you can simply run all tables through your Hercules database, or split out the hat.sql file into its own database, whichever you prefer. If you foresee server expansion, keeping your HAT tables away from your Hercules database would be advantageous.

If you have more than one RO server, you need to figure out how they are laid out. As a rule of thumb, if you have a database that the login table is filled on, you must run hat\_login.sql against that database. If you have a database with characters, items, guilds and parties, you must run hat\_charmap.sql against that database.

## System Performance Monitoring

HAT has the ability to monitor the performance of the server running Hercules. It does this via a python script with the psutils package.

The python script in ~/assets/psutil is run on the remote server and the results of the script are put into an XML file. The XML file is copied back to the HAT server, then parsed by HAT for display.

To make use of this feature, you must have python and python-dev/python-devel installed on the remote server, at a version 2.7 or greater (most major Linux distributions come with python 2.6+ now-a-days) but you must install python-dev (debian like) or python-devel (RHEL like) separately. Additionally, you must have python-lxml installed using the package manager of your choice. You will have to use pip to install the psutils and lxml package. Pip is shipped with python 2.7.9+ and 3.4+. If you do not have these versions or find that pip is not installed, follow the instructions here: <https://pip.pypa.io/en/latest/installing/>

After pip has been installed, you should be able to execute the following command on the Hercules server. You must either be root (or use sudo), or if this is not possible, you can install for your user only. Which means you **must** install psutils with the user account you use to start the Hercules server.

pip install psutil

(or if root is not available):

pip install -–user psutil

Once pip, python-lxml and psutil is installed, you are good for the Hercules remote server.

For the HAT server, you will need php-simplexml. You may need to install via composer. Check your distribution of choice to install simplexml for your PHP.

# Configuration

The configuration of HAT is all done in /application/config.

## Config.php

This file contains most of the basic settings, including paths and cookie settings. Whatever isn’t listed you shouldn’t (have to) change.

**$config['base\_url']** Enter the full base URL (including the http://) of the panel. Make sure you include the trailing slash!

**$config['encryption\_key']** Enter in an encryption key, a random jumble of letters and numbers. You should never need it again, so just make something up

**$config['sess\_cookie\_name']** This is usually set to something like `admin\_yourro\_session` to identify it properly.

## Hat.php

This file contains a lot of the configuration options for HAT specifically. You will need to change most of the settings in here.

**$config['panelname']** If you want to change the name of the panel, you can do so here. Default is `HercAdminTool`.

**$config['servername']** Definitely configure this to your server name, so that your admins can identify your panel comes from your server.

**$config['emailfrom']** the “from” field of email addresses sent by HAT. This doesn’t need to be a “real” email address but you should have permission to send through the domain you use (so don’t use [support@google.com](mailto:support@google.com) or something).

**$config['inactive\_time']** The amount of time in minutes we render an admin user inactive and remove them from the “who’s online” list, if they don’t log out manually.

**$config['hat\_path']** Full path of your HAT installation, including forward and trailing slash.

### HAT multi-server configurations

HAT has the ability to have multiple configurations of login/char/map servers. You can have 1 login server controlling 2 sets of char/map servers and 1 more login/char/map all on the same server, or any combination thereof (so for example, a hi-rate and low-rate production server, then a test server with its own login/char/map servers and databases).

### Login servers

The array **$config['login\_servers']** contains the connection and options for your Hercules login server(s). If you only have 1, you can leave the second commented out and only worry about the first one.

**login\_database\_group** indicates what database connection group in *database.php* this server uses to find the login related tables.

**login\_ip** is the IP address that the Login server listens for connections on from your players.

**login\_port** is the port number the login server listens for connections on form your players.

**login\_ssh\_ip** is the IP address of how you SSH into your server. Can be an IP address or hostname.

**login\_ssh\_port** is the port number of your SSH server at the above IP. Default is 22 but can be different, so change if necessary.

**login\_ssh\_method** is how you login to your SSH server. SSH servers can be configured to accept auth from a **plain** method (entering in a username and password) or a public/private **key** method. Either selections are valid here.

**login\_ssh\_user** is the username you login to the SSH server. This must always be given whether your method is plain or key.

**login\_ssh\_pass** is the password you use to login to the SSH server with the given username. Will be ignored if the method is set to `key`.

**login\_ssh\_pubkeyfile** is where on your webserver your public/private key is located. For security reasons, this should be anywhere on your server except where your webfiles are hosted from so that there’s no chance of it being compromised.

**login\_ssh\_prikeyfile** same as above except this is the private key.

**login\_server\_path** the path to find your login server files and executable. Full path, include forward slash!

**login\_server\_exec** the name of the login server executable. Default is `login-server`.

**last\_mac\_addon** if your server uses the Last MAC Addon, you can turn this on or off here.

Each login server can be defined by making a new key in the `login\_servers` array. Each key should be unique – don’t make a login server with two keys the same or you will error the panel out (and could corrupt your data!).

**$config['default\_server\_id']** is the server ID that you will direct your newly logged in admins to.

!!! Note about default server ID !!!

In the panel you can grant or revoke access to certain servers for your GM’s. So for example, you can have a developer group able to access server ID 10, which is your test/development server. ID 1 is your production server. If you set `default\_server\_id` to 10, your users will get access to this server even if they aren’t supposed to!

### Ragnarok Servers

The array **$config['ragnarok\_servers']** allows you to setup multiple sets of char/map servers for your panel to administrate. Each server gets an ID. There is no restriction on the ID being sequential, however you must have at least 1 server, and it must start at ID#1. Note that HAT (as well as Hercules) doesn’t support multiple map servers, only multiple char/map server sets. Note that the char and map server must be running on the same physical server.

**Servername** is the human readable server name for this char/map server, and what will be selectable in the server selection dropdown for your admins.

**map\_servername** is the name you gave the server in your login table instead of “s1”.

**main\_database\_group** indicates what database connection group in *database.php* this server uses to find the char/map related tables.

**log\_database\_group** indicates what database connection group in *database.php* this server uses to find the log tables. This can be the same/different than the **main\_database\_group** setting.

**server\_ip** is the IP address that the char/map server listens for connections on.

**login\_server\_group** is the group ID of the login server for this char/map server set, set above. Note you can have multiple servers use the same login servers, or your server use all different ones.

**char\_port** is the port number that the character server uses.

**map\_port** is the port number that the map server uses.

**server\_ssh\_ip** is the IP address of how you SSH into your server. Can be an IP address or hostname.

**server\_ssh\_port** is the port number of your SSH server at the above IP. Default is 22 but can be different, so change if necessary.

**server\_ssh\_method** is how you login to your SSH server. SSH servers can be configured to accept auth from a **plain** method (entering in a username and password) or a public/private **key** method. Either selections are valid here.

**server\_ssh\_user** is the username you login to the SSH server. This must always be given whether your method is plain or key.

**server\_ssh\_pass** is the password you use to login to the SSH server with the given username. Will be ignored if the method is set to `key`.

**server\_ssh\_pubkeyfile** is where on your webserver your public/private key is located. For security reasons, this should be anywhere on your server except where your webfiles are hosted from so that there’s no chance of it being compromised.

**server\_ssh\_prikeyfile** same as above except this is the private key.

**server\_path** is the path to find your char/map server files and executable. Full path, include forward slash!

**char\_server\_exec** is the executable for your char-server

**map\_server\_exec** is the executable of your map server.

**reset\_map** is the map name to reset character positions to.

**reset\_x** is the X coordinate to reset character positions to.

**reset\_y** is the Y coordinate to reset character positions to.

**showsysinfo** shows the system performance data using third party scripts ran remotely on the Hercules server. See the section `System Performance Monitoring` below.

**$config["ban\_reasons"]** is a list of ban reasons you can pre-set for your GM’s. You can add more by following the convention of increment a number and placing the next in quotes below the rest. You can edit existing as well.

The rest of this file should never need to be changed unless you’ve added custom classes, custom item types, custom item equip locations or a module to HAT.

### Database.php

This file is pretty self explanatory. Each array has connection information for the applicable database(s). You can either upload your hat tables to the same database or a different database than your ragnarok databases. If you have more than one database, you will need to clone the existing settings and create another set. Make sure you name the array $db[‘<name>’] the same as you have in hat.php under the ‘database\_group’ settings.

Do not change the $db[‘hat’] name, but edit the settings accordingly is OK.

# Additional Security Measures

You can take a couple of additional steps to ensure your panel is safe from prying eyes.

**1. Use SSL**

This is probably one of the best things you can do for the security of your server. Obtain an SSL certificate and install it for your panel. A free one won't offer the level of encryption that you might see with taking credit cards, but it's still encryption.

**2. Change the port that Apache listens on**

If the panel is the only thing web related running on the server, then changing the port that the apache server listens on to something a bit more inconspicuous instead of port 80 can be a bit of a life saver. Thusly, it's one step harder for one to find an apache server listening on that server if it's not on port 80.

**3. Hide the panel behind an htaccess password requirement that only your GM's know**

This is almost a must. Use an htaccess password requirement to require sort of a “two step authorization” for your server. Everyone would have the same htaccess information, then their individual logins. You would need to know both to get in.

# Modules and basic usage of the panel

The panel is broken down into “Modules”. There are currently 8 such modules:

\* Dashboard

\* Admin

\* Account

\* Character

\* Guild

\* User

\* Server

\* Ticket

\* Database

\* Gamelogs

Each module getting an entry/submenu on the left with the exception of the 'User' module, this is explained later. The URL structure is relatively easy to follow:

[**http://yourdomain.com/*MODULE*/FUNCTION**](http://yourdomain.com/MODULE/FUNCTION/ENTRY)**/ENTRY**

The Database and Gamelogs modules behaves a bit differently. As you need to have a subfunction for the itemdb or mobdb, or the different logs, the separator between the two is an underscore. So, to access itemdb ID 1233 for editing/viewing, you would need to go to http://yourdomain.com/database/itemdb\_details/1233. To access the search for the atcmd logs, you would need to go to http://yourdomain.com/gamelogs/atcmd\_search

The functions and/or sub functions for each module are pretty much the same and self explanatory across all the modules:

**/module/search = allows you to search that module.  
/module/details/##### = Allows you to view the details of that ID under that module.  
 /character/details/XXXXXX -> X’s = Character ID  
 /account/details/XXXXXXXX -> X’s = Account ID  
 /guild/details/XXXX -> X’s = Guild ID**

So, you search for accounts, navigate to <http://yourdomain.com/account/search>.

The Username is also a drop down to allow the user to change their settings or logout of the panel. Logging out will destroy the session and cookie and require the user to re-login on their next visit.

If you do not grant access to a function for a group, then it will not show up on the sidebar to select and an access denied message will appear if it is directly accessed. Any function which requires a button press and a group is not able to do will be greyed out and disabled.

## First Time logging in

When you uploaded your hat.sql file into your database, your first user was created. The details are as follows:

**Username:** admin  
**Password:** changeme1!

It is recommended you change the password and the username, for obvious reasons. See your Settings to get your email and password changed. You can change your username with the Admin module.

**Protip:** Keep the admin user with an extremely strong password that only you know. Create a group with the most permissions besides the ones that you don’t want to use, then create your username in that group so you will stay away from things you don’t want to use.

# Server Selection

At the top right of the page near your login name you will notice a drop down of the server name with a server icon next to it. This is your server select. If your HAT installation is administering more than one char/map server pair, this is where you select your server. The character, dashboard, server info, guild, item and log modules all depend on the proper server being selected. Once you select a new server, the confirmation page will send you back to the page you were previously viewing.

# Dashboard/Server Info Module

The dashboard offers a quick glance at how your server is doing and what's going on. This page shows you the active admins and what they're doing/viewing, some basic server stats, admin announcements and some graphs showing account registrations, donation stats (if you have it enabled).

The server performance page take a bit to load but gives you a glance at how the server is performing. It includes RAM usage, swap Usage, network stats, and MySQL performance metrics. It uses an app called “Linfo” to obtain json dumps of this metric data and puts it into a nice to view interface. Such information includes RAM usage, CPU load, Hard Drive usage, as well as Hercules performance functions, like the amount of errors or warnings received from the server console.

# Account Module

The Account module allows you to view, edit and delete account information, including storage and Account flags.

## Searching

The search page allows you to search by Account ID, Account name, Email address, Account Gender, Ban State, or GM permissions. Each field is wildcarded, meaning if you search for “y” in the Account Name field, it will show you any account name with the letter “y” anywhere in the name. You may bypass this by including a “=” character BEFORE the account name you wish to search. This will perform a search specific to what you entered, meaning that if you search for “=hello”, the account name must be exactly “hello”.

Searching with the “Is GM?” checkbox will simply evaluate all the accounts on the server for GM level > 0 and display them.

If you wish to search and list all accounts on the server, simply leave all fields blank and click “search”.

## Account List

The account list is produced after you search. Accessing this page directly will be like you searched for “all records”. Each account flagged as GM will display in red text. The default sort is by Account ID, in ascending order. You may click the column heading to sort by that field. The options area to the right allows you to delete an account if you have permissions to do so.

## Account Basic Info

The Basic Info page of the account module allows you to view information about the account. The Account ID, Account Name, Birthdate, Gender, Group ID, Ban information and loginlog information is shown here. Fields which cannot be edited by you will appear greyed out and uneditable (“disabled”). The options area of this screen will allow you to reset the password, PIN, Send Email, Kick Offline or delete the account. Kick Offline will search for any character on that account that is online and remove them from the server.

## Account Blocks

Account Blocks is where you ban the account.

!! Note about Account Blocking !!

The HAT has a very detailed ban account system which allows you to log bans, assign reasons and in future updates, will send the user an email explaining why they were banned and when the ban will expire. It also kicks the account offline should it be online when you ban. Therefore, it is highly recommended to have your GM’s use this feature of HAT *instead of* the @ban/@unban command in game. This will allow you to log the bans so that all GM’s can see them.

Bans which are still active or have past will show in a list here, sorted by block date, newest to oldest. You may also remove a ban which has not expired by using the Unban button under “Options” if you have the permissions. Clicking the button under “Block Comment” or “Unblock Comment” will fold down a text explaining the GM’s comments as to why the ban/unban was done.

## Notes

The Notes page simply allows your GM’s to keep notes about an account. This can be used to log reimbursements, ticket issues, account issues, really anything your heart desires. Notes are sorted by date, newest to oldest. Shown is the GM who left the note and a timestamp when the note was left. Notes can never be deleted.

## Register DB

The Register DB page views the information from the acc\_reg\_num\_db & acc\_reg\_str\_db tables. You may also add, edit or delete this information.

## Global Register DB

This is the same as above, except the global\_acc\_reg\_num\_db & global\_acc\_reg\_str\_db tables.

## Storage

The Storage tab allows you to view, edit or delete items in the account’s storage. Note that account storage editing will not be possible while any character from that account is online. This includes deletion of items in storage. Clicking the “+” under “Details” will show details about that item. If the item is an armor or weapon, the cards will be shown and can be edited or set to “0” to remove. You may also toggle the bound or broken property for the item.

## History

The History tab shows an edit history for the account. This like group, email, password, and birthdate changes are shown here. The table is sorted by Date, newest to oldest. Since it is history, for obvious reasons it cannot be changed.

## @cmd Log

If the account is tagged as a GM, this tab will show up. This tab will display all of the @commands the account has done over all of the characters on the account.

## Characters

This tab shows the characters on the account. You may click right from this page under “Options” to Reset the position of the character if you have permission. Clicking the Character ID will take you to the character details page for that character.

# Admin module

The admin module is the backend module for the administration of the HAT. You will only want to give your Admins access to much of this area due to the problems they can create by moving GM’s around, switching groups and giving GM’s more permissions on the panel than what they may have in game.

## User Management

User Management allows you to add/delete/edit/disable admins on the panel. For the scope of this guide, all people who have access to the panel are considered “admins”. Each admin on the panel is assigned a unique ID, starting at 2000. This ID may not be changed.

Each admin must have a private email listed in the panel. This is where they will receive notifications about activity levels and if their group allows, notifications about activity on the panel every day.

!! Note about deleting panel users !!

It is not recommended to delete panel admins which have had activity under their logins. Doing this will remove the reference to their name and instead show an ID where they had committed an action. For example, if Sally edited an email of a user long ago and her panel admin account was deleted, instead of showing the user “Sally” changed the email, it would show user “2008” changed the email. Instead it is recommended to DISABLE users rather than deleting.

Admins may have their accounts disabled. This will prevent them from logging in to the panel and instead show an error “This user account is not authorized to login. Contact an administrator.” All user accounts initially created will be disabled by default and must be enabled manually after creation to be able to login.

You may link an admin on the panel to an account ID in game. This will allow you to get to that account easily by looking up the admin in the panel, in the event you need to check logs or history on that admin’s in-game account.

“Generate New Password?” checkbox, when checked, will generate a new 15 character random password and immediately email it to the admin’s private email after clicking the “Process & Save Changes” button. You can also do such to all admins by clicking the “Reset All Passwords” button on the User Management list. You may also Disable/Enable All Users except your own (if you have the permissions) by clicking the appropriate button.

## Group Management

??? Note about groups ???

Please remember that Panel group edits **do not** affect in game groups. These are two separate things.

The Group Management interface allows you to group your admins so that similar level admins have the same permissions without needing to change permissions for multiple users. 99 group levels can be set, with id 99 being reserved and not-editable you may setup 98 custom levels for your use. Each group can be setup with multiple permission settings affecting all aspects of HAT. See “Appendix A” for the full list.

A checkbox gives that group permissions to do that action, no checkbox means they cannot make that action.

# Character Module

The character module shows information and allows edits on any aspect of a character.

## Searching

Similar to the account module, you may search characters on the server by many different fields. Searching by Character name or character ID is wildcarded. Meaning that searching for a Character Name of “t” will find any character with the letter “t” anywhere in the name. You may do an exact search by using the equals sign (=) before the query. So, searching for “=this” will look specifically for a character with the name “this”. Base Level and Job level searches are basically Less than or equal to X and/or greater than or equal to Y. To search for characters between two levels, you would put the HIGHER number in the less than or equal to box and the LOWER number in the greater than or equal to box. Like the account search, leaving all fields blank will produce a list of all characters on the server.

## Character List

The character list will be produced when you search. Accessing this page directly will be like you searched for all records. Here you can see a summary of the characters you searched for. On the left shows the character ID, click on it to be taken to the details page for that character. On the right you may reset the position of the character using the button under “Options”. Any character which has a deletion pending (either via admin or user) will have the words “DELETION PENDING” written in the options column and the Reset Position button will be disabled.

## Character Details

The Basic Info tab will show you mostly the stat information about that character. Permissions to edit, if given, will be reflected in the fields able to edit. The fields which you do not have permissions for will be disabled. Midway down the page you have an “Options” section with the ability to Reset Position, Kick Offline or Delete Character. If the character is online, the text “Character is online and cannot be edited!” will appear at the top and the submit changes button will fail when pressed. Note that the Online Status is requeried after submission of the information.

## Items

The items tab allows you to edit or delete any item in the character’s possession (not in storage). There are three sections from top to bottom: Equipped, In Inventory and In Cart. The Equipped section lists all of the character’s equipment, sorted by “Equipped to” in top to bottom, left to right order as it would appear in game. You may click the “+” button next to any item to expand details, such as the refine level, bound status and cards embedded, if an armor or weapon. These may also be changed if you have the proper permissions. Items can be edited by changing the appropriate fields on that item and clicking the “Edit” button on that item’s row under “Options”. Any equipment can also be forcefully unequipped and thrown back into the inventory by clicking the “Unequip” button (handy for clearing appearance crashes). You may also simply delete by clicking the “Delete” button.

## History

The history tab shows you the charlog table on the left, including when the character was created, deleted and logged in. The Character Edit History on the right gives you a log of what edits were done by admins to that character, and what the old and new values were.

## Friends

The friends tab just gives you a list of that character’s friends, where you can jump to their character details or account details.

## Pet/Homun Info

Gives you a breakdown of the Homunculus the character currently possesses.

## Skills

Gives you a breakdown of the server saved skills and hotbars the character has.

# Database Module

The Database module allows you to view, modify and delete items, mobs, skills and cashshop items from your server. If you intend to make use of the Database module, make sure you and your GM’s read this section completely and understand it.

!! Important information about managing databases follows !!

Please make sure you read the following sections carefully and completely before you or your GM’s attempt to manage the item or mob DB’s on your server through HAT. Make sure you always take backups. One wrong move when managing your databases could end very badly.

## Setup

HAT is capable of managing the flatfile db’s. When a .conf file is changed, it will db2sql that file back to the SQL databases for speed in just viewing items on a character/account instead of querying the flatfile everytime. Therefore it is important that you **never** edit directly the .conf files **unless** you run the db2sql script separately. If you edit the .conf file manually your SQL and actual in-game .conf database will be out of sync. HAT does have under Database management a method to run the db2sql script with just a simple click of a button. This is explained a bit later.

# Server Setup

The Server/Server Setup module allows you to change various aspects of the server, including configurations and maintenance.

## Hercules Info/Maintenance

This page combined the server maintenance controls as well as some statistics and server console access. In the top left box the buttons “Login”, “char” and “Map” will show you the server status, whether the server is listening on this port or not. Red means offline and green means online. You can push each button to toggle that status. If the server is down, it will attempt to be started. If it’s up, it will attempt to be stopped. HAT will check up to 5 times, 2 seconds apart for the server status to change. If no change is detected, the panel will alert you that the server failed to start/stop.

You can also use the “Start Server” and “Stop Server” and “Restart Server” buttons to control all of the servers at once. “Start Server” will attempt to start any servers that aren’t running. “Stop Server” will stop servers that are running. “Restart Server” runs the script for “Stop” and “Start” in the same button push.

You can also use the buttons below these to Reload the Scripts, Battle.conf, or @cmd databases.

# Appendix A – Group Permissions

Below is a list of permissions and what each means and controls.

## Account Module Permissions

* View Accounts
  + Can view account information (See note 1 & 2 below)
* View Email Address
  + View Access to email address. Disabled means email will show up as \*@\* for that group.
* Edit Account Email Address
  + Edit the account email address. Disabled will disable the field, making it uneditable for that group. Note enabling this permission will automatically enable “View Email Address” above.
* Reset Account Password
  + Allows the use of the button to reset account password. Disabled will disable the button, making it unclickable.
* Edit Account Gender
  + Disables/Enables the radio buttons to change the gender of the account.
* Add Game Account
  + Allows the users in the group to add a new account to the server, bypassing email/admin authorization. Such an account would be immediately usable.
* Edit Game Account Group ID
  + In conjunction with below, allows the user to edit the Group ID.
* Max group level able to edit
  + Controls what group level in game the user is allowed to do any edits on. This also controls how high the user is able to set the Group ID. If the game account is higher level than this setting, the user is unable to edit any part of the account, even if they have other permissions to do so. This includes Storage Items, Account flags, characters on the account and bans.
* Edit Account Birthdate
  + Allows admin to edit account birthdate. Disabled will disable the field, making it uneditable.
* Edit Account Reserved Slots
  + Allows the admin to edit the reserved slots for the account. Disabled will disable the field, making it uneditable.
* Purge Inactive Accounts
  + NOTE: Critical permission. Allows admin to use purge function to remove inactive accounts older than # days (user changeable). Inactive accounts are immediately removed, along with the items, characters and guild memberships.
* Ban Account
  + Allows admin to place a ban on an account. Such ban can be for any duration and for any reason.
* Unban account
  + Allows admin to remove any ban on an account.
* Edit Account Trust
  + Currently unused.
* Edit Item In Account Storage
  + Admin will be able to edit or delete any item in the account’s storage.

NOTE 1: All admin accounts that can view game accounts can see the following information:

* Account ID
* Account Name
* Last Login
* Create Date
* Account ban Information
* Gender
* Group ID
* Birthdate
* Last IP
* Login Count
* Notes
* Account Register DB
* Account History
* Account Storage
* Account Characters

NOTE 2: All admin accounts that can view game accounts can edit/use the following information:

* Account Notes
* Send Email
* Kick Offline (Only if “Kick Character from server” permission is given in Character Module)

## Character Module Permissions

* View Who’s Online
  + Who’s Online is available to the admin. This is active whether or not WoE is active and is expanded info showing the levels, zeny and last location of all people listed as online.
* Edit Character Name
  + This will allow the admin to change the character name.
* Edit Character Slot
  + This will allow the admin to rearrange the character positions on the account.
* Edit Character Zeny
  + This will allow the admin to edit the zeny of any character directly.
* Edit Character Levels
  + This will allow the admin to edit the base/job level of any character directly.
* Edit Character Stats
  + This will allow the admin to edit the STR, VIT, AGI, DEX, VIT, LUK, Stat Points or Skill points of any character directly.
* Change Character Job
  + This will allow the admin to change the character class to anything available.
* Change Character Look
  + This will allow the admin to change the hairstyle, hair color or clothes color of any character.
* Delete any character item
  + This will allow the admin to delete any character item.
* Edit any character item
  + This will allow the admin to edit any character item.
* Send Item via mail
  + Currently unused.
* Kick Character from server
  + This will allow the admin to kick a character from a server through the panel
* Delete individual character
  + This will allow the admin to delete the character and anything associated with the character from the server. Read more in the character module information in the guide for what this entails.
* Restore Deleted character
  + This will allow the admin to restore a deleted character
* Reset character position
  + This will allow the admin to use the “Reset position” button to move the character back to the main town.

## Admin Panel Module Permissions