

Area Protection Rising World plug-in version 1.0.4

by

Maurizio M. Gavioli
(a.k.a. Miwarre)

© Copyright 2018, Maurizio M. Gavioli, licensed under the [Gnu General Public Licence v. 3](#)

Note:

The images of this manual have been prepared while the plug-in was in development and they may differ slightly from the released version.

Server / World Owner

This part of the manual interests server owners more than players. It covers installing the plug-in, configuring it and importing data from the old LUA area protection script.

Installation

Expand the ZIP file into the `plugins` folder of the main Rising World folder, obtaining a folder structure like:

```
<RisingWorld>
├── plugins
│   ├── ap
│   ├── locale/
│   ├── preset/
│   ├── resources/
│   ├── ap.jar
│   ├── AreaProtection_Manual_jpg.pdf
│   ├── COPYING
│   └── settings.properties
```

If the server has the old LUA script for area protection installed:

- you may want to import the areas defined with it: for this, see the “*Importing areas from the old LUA script*” section below;
- **remove the LUA script: the old script and the new plug-in cannot go together!**

Finally, restart the Rising World server.

Configuration

The `settings.properties` file contains the configuration settings which can be customised. Each setting is commented: refer to the file itself for details. **Please revise each setting in this file to check if the default suits your server or not.**

locale/

The `locale/` folder contains the files to internationalise the UI texts. A `readme.txt` file in it explains how to use this feature.

presets/

The `presets/` folder contains the definitions of some 'common' sets of player rights, i.e. the sets of rights a server owner expects to use more frequently, in a convenient, ready-to-use way. However, actual player rights are not limited to these sets, as the plug-in allows to define specific rights for each player, if needed.

There can be any number of presets; they take memory on the server, though, so it is better to stick to the ones really useful.

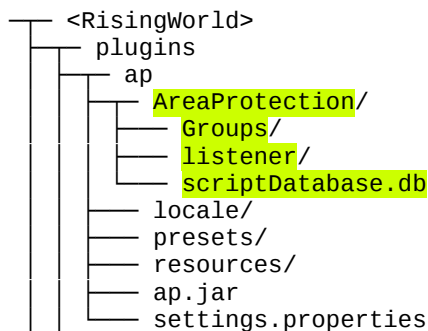
They use a syntax very similar to the the syntax used for the `Groups` of the old LUA script, so that these old Groups can be directly moved into the `presets/` folder (and renamed from `.group` to `.preset`) replacing the default ones coming with the plug-in itself.

A few of the old keywords are not supported (but accepted) and some new are added, to suit the new permission system of the Java plug-in API: the `presets/admin.preset` file details all these changes.

Importing areas from the old LUA script

Areas and player rights defined with the old LUA script can be imported into this plug-in without recreating them from scratch.

Before re-starting the server with the new plug-in installed, move the **entire** `AreaProtection` folder of the old LUA script with its own `Groups` sub-folder into the `ap` own folder created during the installation:



The folder must be named `AreaProtection`, respecting the capitalisation and **without a blank space** between the two words.

Then start the Rising World server: the areas and the player rights defined in the old script will be imported.

The process may take several minutes: do not worry if the server takes longer than usually to complete the start-up. This is a once-only procedure: the next server start-ups will be normal.

Note: Chests data are NOT imported.

After that, you can remove the `AreaProtection` sub-folder or leave it in place, as it will be no longer used by the plug-in.

In any case, remember to remove the old LUA script from the `scripts` folder: **the old script and the new plug-in cannot coexist.**

Recycling LUA script Groups

Groups from the old LUA scrip can be used almost immediately by moving them to the `<RisingWorld>/ap/presets/` folder; note anyway:

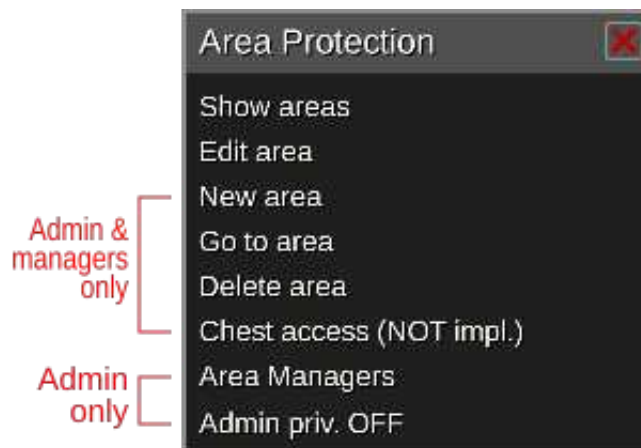
- group files have to be renamed from `*.group` to `*.preset` ;
- you may want to revise them and, for instance, add to them the new permissions.

User Manual

This part of the manual describe the plug-in operations.

Chat Command

There is one main chat command: `/ap` (press `T` to open the chat); it brings this menu up:



- `Show/Hide areas` toggles on/off the display of the defined areas.
- `Edit area` allows to edit any area owned by the player. Server admins and managers (see below) own all areas, other players own only the areas they have been made owners of by an admin.
- `New area` (admins/managers only) starts the creation of a new area.
- `Go to area` (admins/managers only) to jump to a specific area.
- `Delete area` (admins/managers only) deletes an existing area.
- `Chest access` (admins/managers only) manages access to individual chests (not implemented yet).
- `Area Managers` (admins only) adds and removes area managers, i.e. regular, non-admin, players who can work on areas as admins.
- `Admin priv. OFF` (admins only) controls whether admins are subject to the same area permissions as regular users, without any privileged access to areas; useful for local debugging or testing of settings, when no regular users are available; in any case, admins still keep the privilege to create new areas, but for area editing, adding players to areas, player editing and allowed operations within protected areas, admins are considered regular users.

Two additional chat commands have been added, even if they duplicate a function already covered by the main menu, as players may expect them to work:

`/showareas` turns on the display of existing protected areas for that player;

`/hideareas` turns off the display of existing protected areas for that player.

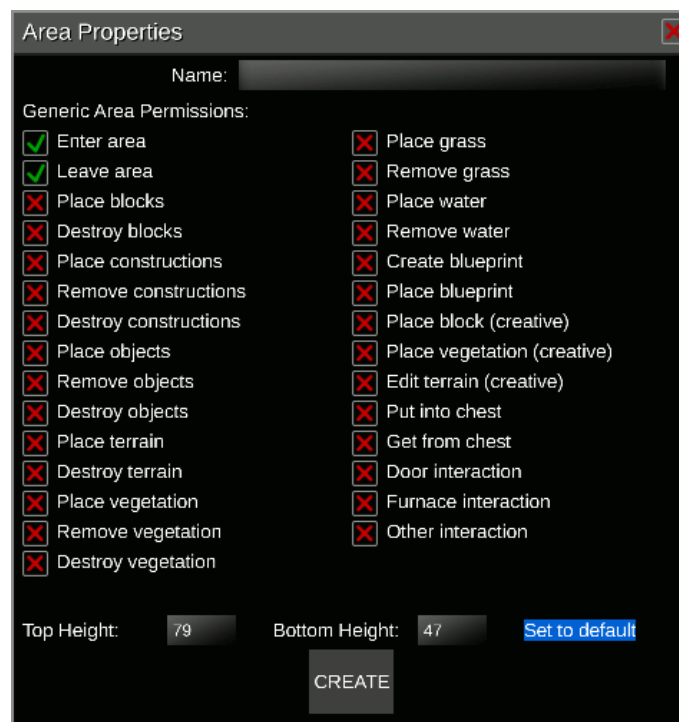
Creating a new area (admin only)

Only server admins (and area managers) can create new areas. To create a new area:

- Bring the main menu up with the `/ap` chat command and select `New area`.
- Follow the on-screen instructions to select the required area; the process is very similar to the old script process. During the area extent selection process, a small window in the left bottom corner displays the centre and the size of the area being selected, to help defining areas of specific extension:



- In this stage, it is not necessary to precisely set the top and bottom height of the area, as they can be adjusted later.
- At any moment, you can press `[RETURN]` to create the area or `[ESCAPE]` to abort the process.
- Pressing `[RETURN]` displays this dialogue box:



- Click in the `Name` text field, type a name for the new area and press `[RETURN]` (or click outside of the text field).
- The area is initially created with the default permission: any player can enter the area and leave it,

but nothing else. Turn on or off the permissions as required.

- Top and bottom boundaries of the area height can be set; the **Set to default** button sets them to default values stored in the **settings.properties** file; this allows to have areas with consistent vertical spans, if necessary.
- Once the default, general permissions of the area are as you want them, press the **CREATE** button and the area will be saved and you'll be back to usual Rising World play.

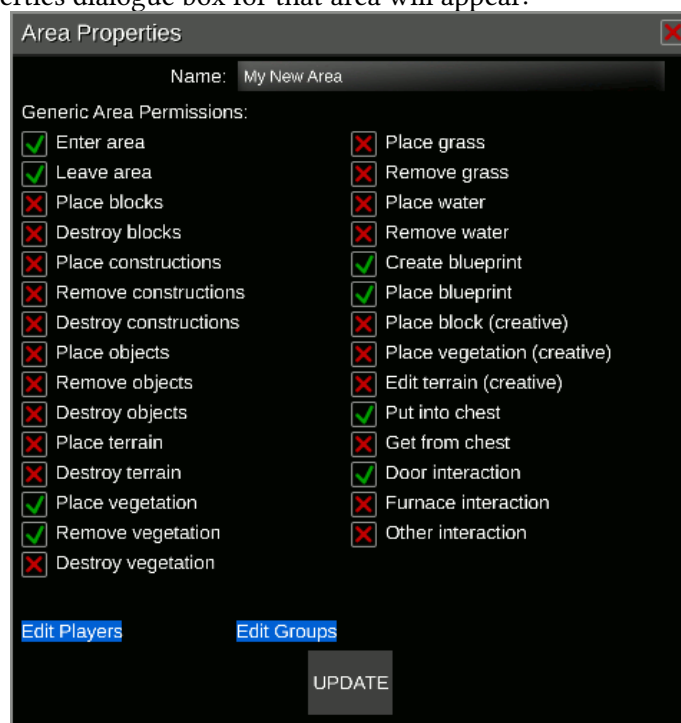
Editing an area (area owners only)

Each player who owns an area (server admins and area managers own all areas) can edit it. Area editing includes:

- changing the general permissions (the permissions applying to everyone without specific permissions);
- adding/removing players with specific permissions
- adding/removing groups with specific permissions

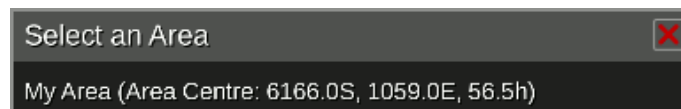
To start editing an area:

1) Move inside the area, bring the main menu up with the **/ap** chat command and select **Edit area** : the Area Properties dialogue box for that area will appear:



OR

2) if you are outside any area, bring the main menu up with the **/ap** chat command and select **Edit area** : a dialogue box listing all the areas you can edit will appear:



Select the area you want to edit and its Area Properties dialogue box will appear.

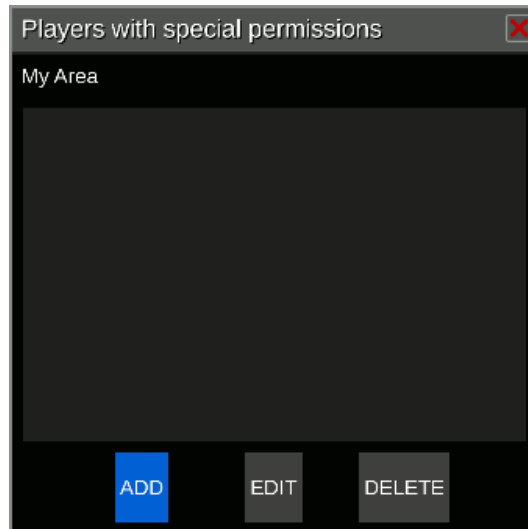
Adding players with specific permissions (area owners only)

Each player who owns an area (server admins and area managers own all areas) can add to the area new players with specific permissions.

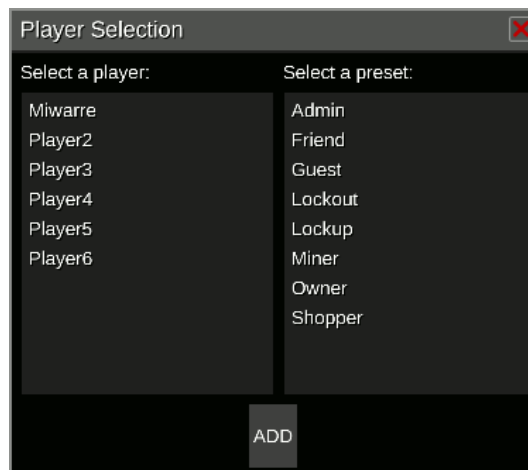
Admins: use this procedure to set the owner(s) of each new area created.

To add one or more players:

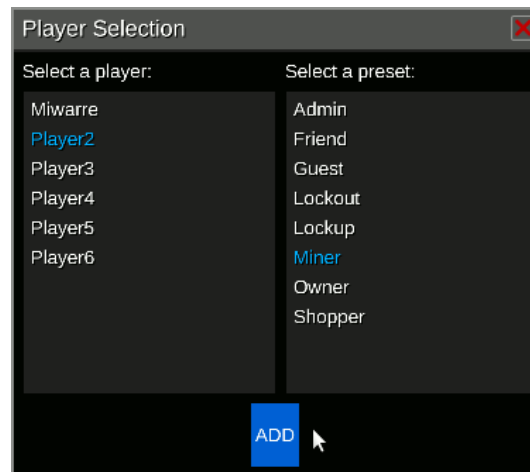
- Start editing the Area Properties as described above.
- To manage players with specific permissions, click on **Edit Players** :



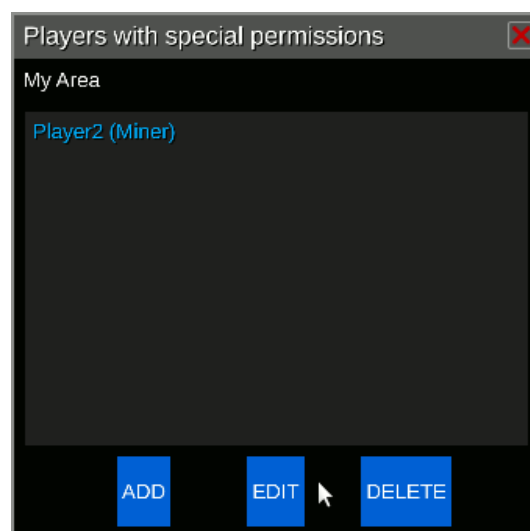
- The area currently has no player added; to add a new player, click on **ADD** :



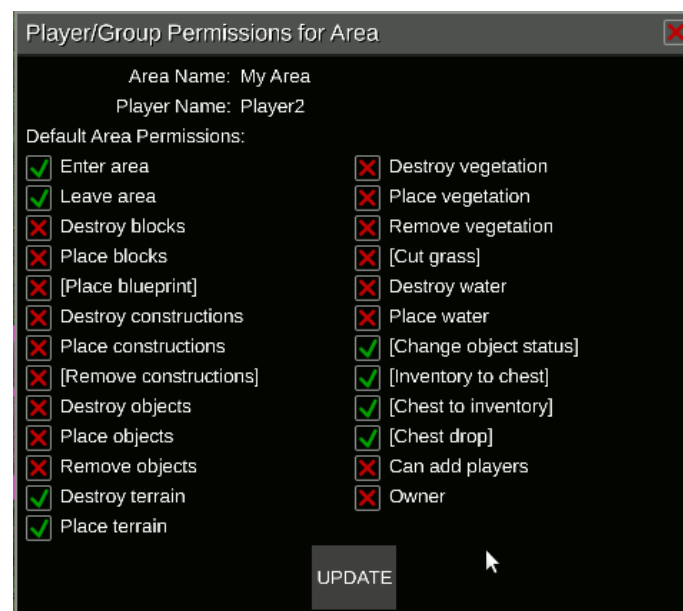
- Select a player **and** a preset (to assign some initial permissions to that player):



- and press **ADD** to return to the player list, with the new player added.
- From there, you can add more players and/or select a player to edit or remove it:



- Editing the area permissions for a player, brings up the player permission dialogue box:



- **Note:** a player who is not a server admin can only edit the permissions which has been granted himself; in other words, in the dialogue box the permissions which he does not has himself are

disabled and he cannot grant them to other players.

- **Note:** The “Owner” right can only be managed by admins; even the owner of an area cannot make another player an owner (unless he is an admin himself).

Adding groups with specific permissions (area owners only)

Each player who owns an area (server admins and area managers own all areas) can add to the area specific permissions for groups. The procedure is almost the same as adding players, except that it is started by clicking the **Edit Groups** button in the Area Properties dialogue box, instead of the **Edit Players** button.

“Groups” refer to the server groups, those defined by the files in the `<server>/permissions/groups/` folder.

Permission priority

The permissions actually given to a player are determined in this way:

- If the player has specific permissions for that area, they are used.
- If the player has no specific permissions for that area, but belongs to a group which has, the group permissions are used.
- If the player group has no specific permissions or the player does not belong to any group, the generic permissions for the area are used.

Permission combining

When a player is inside several overlapping areas, he is granted the *intersection* of the permissions for each area; in other words, he cannot do any action which is forbidden by at least one of the areas he is within, even if the other areas allow it.

Changes to permissions

A few permissions work differently from the old LUA script:

- “Remove (or pick) vegetation” is reserved to collecting fruits (or similar) while leaving the plant in place able to produce new fruits; this includes apples, cherries, lemons, tomatoes, chili, corn and cotton.
- When collecting the plant produce destroys the plant itself (as with watermelons or hemp), the “Destroy vegetation” permission applies, as when cutting trees down.
- Chest access permissions have been reduced to two:
 - “Putting into chests” (currently equivalent to the old “Inventory to chests”, but in future may include new forms of contributing to chest content);
 - “Getting from chests” which includes both the old “Chests to inventory” and “Chest drop”.

Player location vs. Event location

One of the settings in the `settings.properties` file — namely, `eventPos` — allows to select whether to use the permissions active **at the event position or at the player position** to check if an event is permitted or not.

Event position (`eventPos=1`) : this is the default and was how the old LUA script worked too. This setting follows precisely the area edges, avoid any “reach-in effect”, but it is slower.

Player position (`eventPos=0`) : this settings introduces a “reach-in effect” which makes areas slightly smaller than defined; in practice, when a player is very near the border of an area but outside it, for a short range inside the area, he might be able to carry out an action he has no right for if he can reach the point inside the area without physically entering it himself. This setting is faster, though.

Exceptions are:

- the Blueprint events (“Create blueprint” and “Place blueprint”): all the areas intersecting the blueprint extent are checked rather than any single point;
- the chest events (“Put into chest”, “Get from chest”), which contain no event position and for which the player position is checked anyway.