

1 Overview

These are the major features of Community Bridge:

1. Minecraft player to web application user “linking”.
2. **(NOT YET IMPLEMENTED)** Authenticating a user against the web application database.
3. Group synchronization (currently a misnomer, since the “synchronization” is only one way)
4. Recording statistical information about the players in forum custom fields.
5. Kicking of players who are banned from the forum.

The first of the features is essential, without it, none of the other features of the plugin would work. Given that the linking feature’s importance, it is the first thing to evaluate when ensuring that CommunityBridge is operating correctly.

2 Linking Feature

For each of the following configurations, CB should:

- Recognize a player as not being registered, e.g., not present in the web application’s database.
- Correctly identify a player as being registered—identifying the correct user.
- **(NYI)** If authentication is off **DO NOT** force authentication.
- **(NYI)** If authentication is on, handle a valid authentication by...
- **(NYI)** If authentication is on, respond to an failed authentication attempt by...

Possible Configurations (order of complexity):

- Minecraft playername and web application username are required to be the same. Required information: users table, user id column, username column. Configuration Name: same_name.
- Minecraft playername and web application username can be different; the Minecraft playername is in a separate column on the users table. Required information: users table, user id column, minecraft playername column. Configuration Name: diff_name_same_table.

- Minecraft playername and web application username can be different; the Minecraft playername is on a different table in its own column (frequently, this is because it is implemented through a custom profile field in the web application). Required information: tablename, user id column, minecraft playername column. Configuration Name: diff_name_diff_table
- Minecraft playername and web application username can be different; the Minecraft playername is on a different table that data is stored in the key-value format. Required information: tablename, user id column, key column, value column, data key name. Configuration Name: diff_name_diff_table_keyed

2.1 Proposed config.yml Section

```
# Settings associated with linking a minecraft player with a web application's
# user. As this feature is a prerequisite for all other features, it cannot
# be disabled.
player-user-linking:
  # Set this to true if you want to require the player to confirm that they're
  # the player that is registered on the web application by providing their
  # web application password.
  authentication-required: false

  # Set these to true to inform players when they log in if they're linked to
  # the web application. These correspond to the link-unregistered-player and
  # link-registered-player messages in messages.yml.
  notify-unregistered-player: true
  notify-registered-player: true

  # If you want the player disconnected from the game if they haven't
  # registered set this to true.
  kick-unregistered: false

  # This needs to be set to either same-name, same-table, multi-table, or
  # multi-table-with-key.
  # same-name: Use this option when you want the minecraft playername and the
  # web application username to be the same.
  # The remaining three options are for when you want the minecraft playername
  # and web application username to be different.
  # same-table: Use this option when the player's minecraft name on the same
  #               table as the users table in its own column.
  # multi-table: Use this option when the player's minecraft name is on a table
  #               separate from the users table in its own column.
  # multi-table-with-key: Use this option when the player's minecraft name is
  #                       on a table separate from the users table that stores
  #                       data in key-value pairs.

linking-mode: same-name
# Configuration for the table that the web application stores user information
# on.
users-table:
  # The name of the table that the web application stores users on.
  name: users_table
  user-id-column: user_id
  username-column: username
# Configuration for the table that the minecraft player name is stored on by
# the web application.
playername-table:
```

```
# Name of the table that contains the minecraft player name.
name: users_table
# Column containing the user's user id on the table containing the
# minecraft player name.
user-id-column: user_id
playername-column: minecraft_playername
# These three are only used if the player's minecraft name is on a
# table that stores its data in a key-value pair.
playername-key: ''
key-column: ''
value-column: ''
```

2.2 Proposed messages.yml section

Messages related to the linking feature in the messages.yml.

```
link-unregistered-player: Unregistered Account - Please register at YOURFORUMURL for
link-registered-player: Registered Account, Linked to Forums.
link-unregistered-reminder: Just a reminder to visit YOURFORUMURL and register today
```