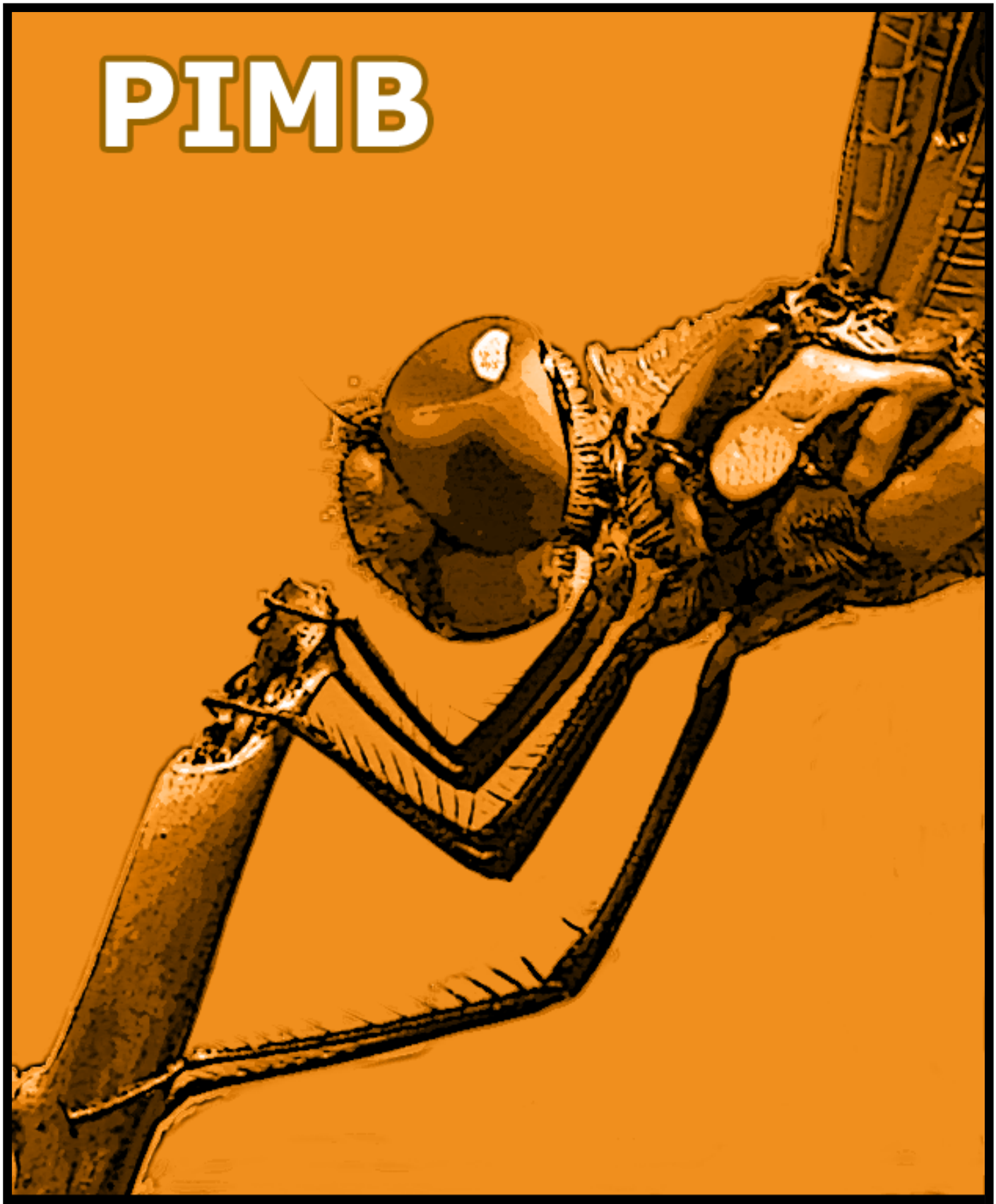
PIMB



CONQUER THE WEB.

PIMB is an open source instant messaging framework that allows you to build, run and monitor chat bots with very little effort.

PIMB Manual

Author: Ferdy Christant
Date: 2008-01-8
Version: 0.60
Status: Final
Reference: S3/PIMB/01_0.60
No. of pages: 17 (incl. This page)

Table of contents

1. Introduction

This manual describes how to install setup and use PIMB v0.60. Note that this software is experimental, making it extra important that you carefully read this manual.

1.1 What is s3maphor3?

S3maphor3 is a small software brand founded by Ferdy Christant in 2004. The brand mission is to deliver fun, innovative, open, honest, high quality products and services to the open source community.

A dedicated site for the brand is available at:

http://www.s3maphor3.org/

1.2 What is PIMB?

PIMB stands for PHP Instant Messaging Bots. It is an open source instant messaging framework. Using PIMB you can easily create and run instant messaging bots that provide useful services. Besides offering a framework for instant messaging bot development (PIMB API), PIMB also runs your bots, and allows you to control and inspect them using a web panel (PIMB Engine). Although simple in concept, PIMB releases a wealth of opportunities. Any logic you can code in PHP, you can include in your bots' service to other IM users.

Throughout this manual you will learn what PIMB is all about. If you would like to see it first, I recommend the following video:

http://s3maphor3.org/pimb/pimbvideo_050.php

Note: The video demonstrates version 0.50 of PIMB, which is similar in usage and capabilities to PIMB 0.60.

Other PIMB information you can find at the dedicated project site:

http://s3maphor3.org/pimb/

1.3 Licensing

Although the PIMB software is free, and the source code is available for free as well, there still are some conditions for using it. By using it, you will agree to:

- Not make any commercial attempt with the software.
- Not discredit the author of the software in any reasonable way.
- Not distribute copies of this software by anything else but the official s3maphor3 site.

You are not required, but urged to:

- Send your feedback as a user of the software to this author.
- Send possible improvements for the software to this author, so everybody can benefit.
- Keep this author updated about your online activity regarding the use of this software.

You are allowed to:

- Use the software for free without a time limit.
- Alter the software, under the conditions mentioned above.
- Deploy multiple copies, under the conditions mentioned above.

In all cases, s3maphor3.org will remain the owner of the software

1.4 Known limitations

The current version of PIMB, v0.60, is an alpha release. Given the early status of the product, you should be aware of at least the following limitations:

- PIMB is developed and tested in a LAMP environment. Technically there should be no reason why it should not work in a Windows environment, but it has not explicitly been tested.
- PIMB bots essentially are PHP scripts that run in a continuous loop, waiting for messages from users. Some deployment situations do not allow you run a PHP script forever, nor do they allow you to change configuration settings that allow it. The PIMB software offers an alternative solution in those cases: it allows you to start bots from the command line.
- Currently, PIMB bots do not automatically authorize all human users that want to make use of the bot service. Instead, authorized users must be manually trusted before they can use the both. This limitation is likely to disappear in the future.

1.5 Support

PIMB is no commercial offering; it is a voluntarily developed product. Therefore, support is based on a best effort basis. In practice this means that all support requests are taken seriously, registered and handled professionally. The s3maphor3.org brand has different ways of requesting support, including a forum, email, and an online ticket system. You can request support at:

http://s3maphor3.org/support/support.php

The forum can be found here:

http://s3maphor3.org/forum/

Please, by all means, help us make PIMB better!

2. Installation & Setup

This chapter will help you get PIMB up and running. It consists of the following steps:

- Download the software
- Upload the software to your web server
- Set write permissions on directories
- Tweaking the configuration file (optional)
- Configuring the test bot
- Testing the installation

Before starting the installation, please consider the following system requirements:

- PIMB requires PHP 5 or higher
- Safe mode needs to be disabled, this will allow a PHP script (bot) to run forever

2.1 Download the software

You can download the PIMB software here:

http://s3maphor3.org/pimb/

Check the "downloads" section of the PIMB homepage for the latest release. After downloading the release, extract the archive file to a local directory on your file system.

2.2 Upload the software to your web server

The next step involves uploading the PIMB software to your web server. The way to upload the files may differ per situation. Most ISPs offer FTP functionality to do this. Make sure you upload the software somewhere inside the public folder of your web server.

Note: PIMB has been developed and tested in a LAMP environment. This alpha release has not explicitly been tested for Windows environments.

2.3 Set write permissions on directories

PIMB needs write permissions on at least the "log" and "locks" directories. Therefore, you need to set the write permissions on those directories to allow this. There are different ways to do this. From the command line interface:

```
chmod 777 log
chmod 777 locks
```

Some ISPs offer a control panel to set these write permissions. Some FTP clients also allow you to set these file permissions.

Note: If your own bots write to the file system, you obviously also need to set the write permissions on the directories your bots write to.

2.4 Tweaking the configuration file

PIMB allows you to override some basic settings in the "pimb.conf.php" file, located in the root directory of PIMB. Optionally, you can tweak these settings to your own needs. They are:

- VERSION: version indicator of the product, there should be no need to change this
- MAX_LOG: the maximum amount of log messages to display on the log page of bots
- DEBUG: when enabled, your bots will log a keep-alive signal every 5 seconds.

2.5 Configuring the test bot

PIMB comes with a test bot, called the Parrot bot. It is an extremely simply bot script that simply replies whatever you say to it. Still, it is a useful bot to explain the process of creating your own bots, and to test if your PIMB installation is working correctly. For now, we will focus on setting up the bot to work, and worry about the details later.

First, we need to register an instant messaging account to associate the bot with. This instant messaging account will be used as the "contact" to add by bot users. This early release of PIMB is based on the Jabber network, therefore we need to register a jabber, or jabber-compatible account. In practice, it is save to register your bot as either a Jabber (@jabber.com) or Google talk account (@gmail). Other compatible networks may work, but are not all tested.

The easiest way to register a bot account is to download and install a Jabber-compatible instant messaging client. A list of clients can be found here:

http://www.jabber.org/software/clients.shtml

PIMB has been tested using the PSI client, which can be downloaded here:

http://psi.affinix.com/

Once you have installed your favourite client, you need to register two accounts. This can be done from inside the client. The first account is your own personal account, which you as a human user will use to interact with the bots. The second account you need to register is the account of the bot. Typically, you will need one account per bot. In my case, I have registered the following accounts:

fchristant@jabber.com (my personal account)
s3_testbot01@jabber.com (the account of my Parrot bot)

Note: You should make up your own unique names for these accounts, and not use the example account names.

The next step to take is to associate the bot account you just registered with the bot file of the Parrot bot. Open up the file "parrot.ini.php", and locate the following settings:

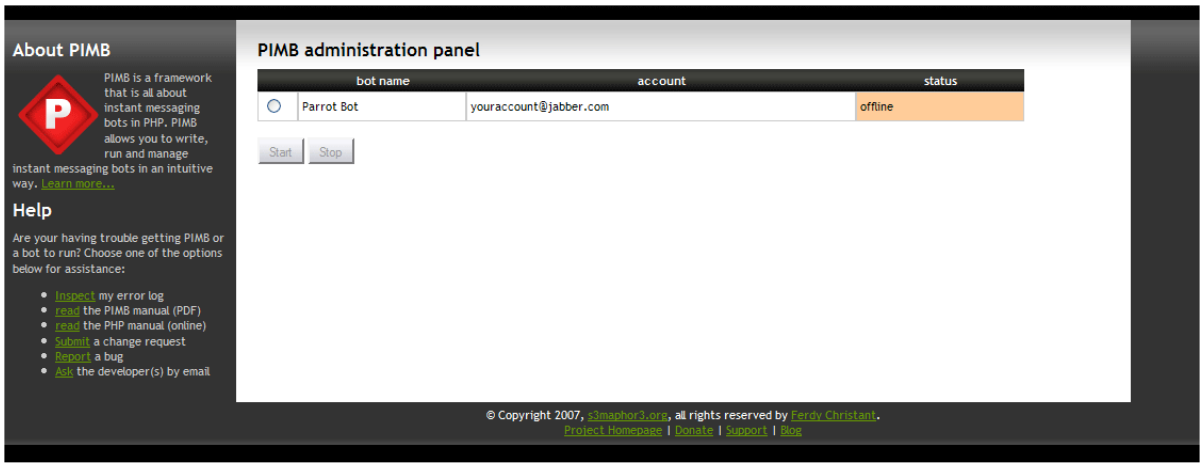
- Server. Set this to the server you registered the account at, i.e. "jabber.com" or "gmail.com"
- Bot_user. Set this to the account name of your Parrot bot, i.e. "s3_testbot01"
- Bot_password. Set this to the password of your Parrot bot account

Save the file. If you followed all the steps above correctly, now it is time to test our installation.

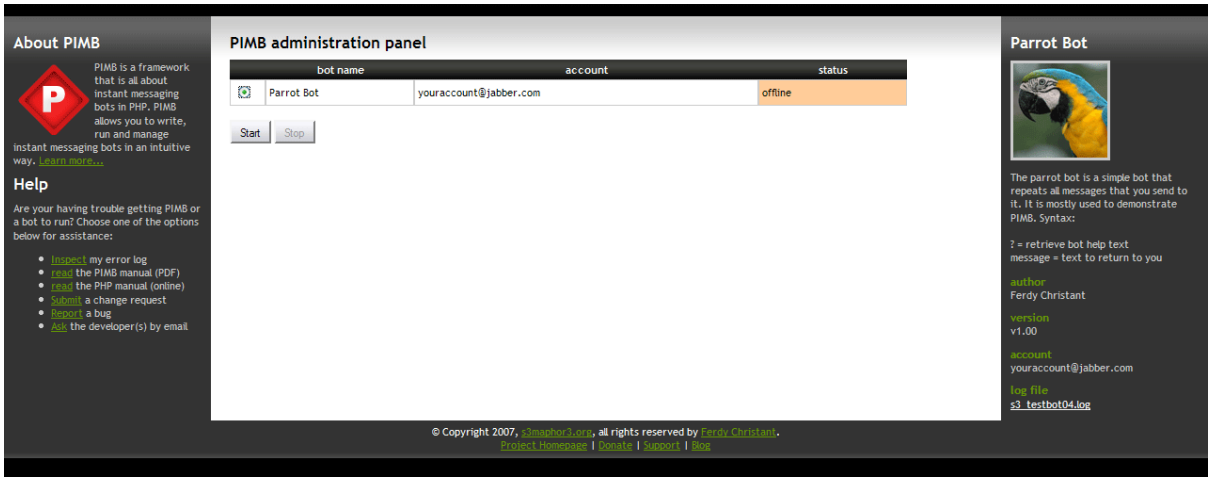
2.6 Testing the installation

To test if our basic PIMB installation is working, follow the steps below.

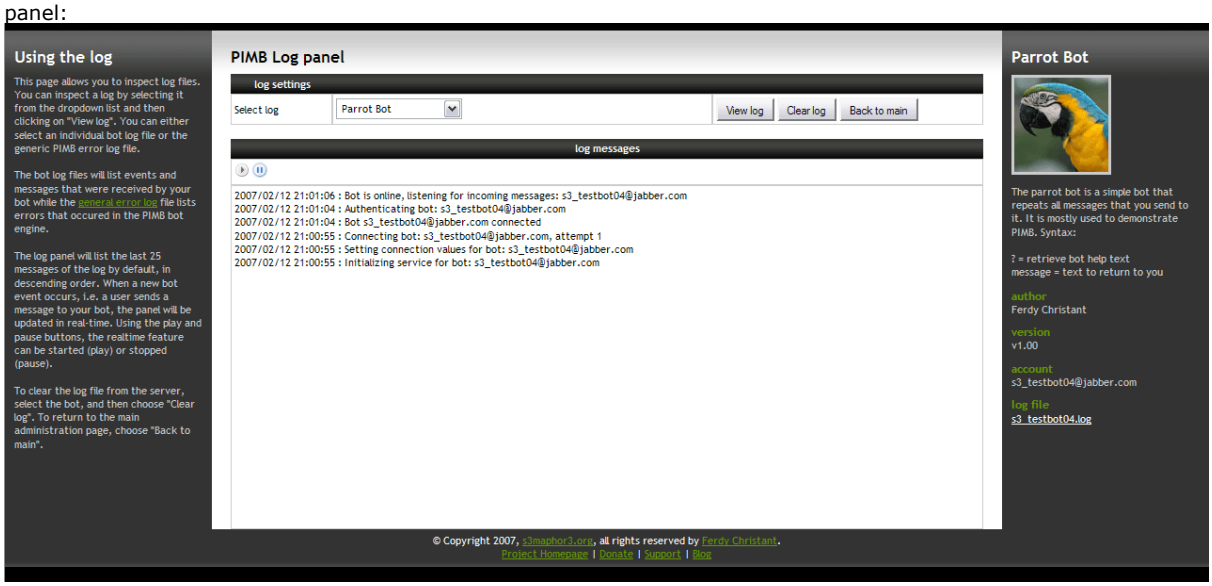
First, open up the PIMB web panel, found at <yourserver/yourpath/PIMB>. It should look something like this:



Next, click the radio button in front of the Parrot bot:



As you can see, this opens up the details of the Parrot bot. Before we try to start the bot, log on to the instant messaging client you installed earlier. Log on using your human user account and add the bot account to your buddy list. Next, return to the PIMB admin panel and click the start button. Allow the bot some time to start up. Once the status indicator displays "online", open up the log file. The link to the log file is located on bottom right of the admin panel:

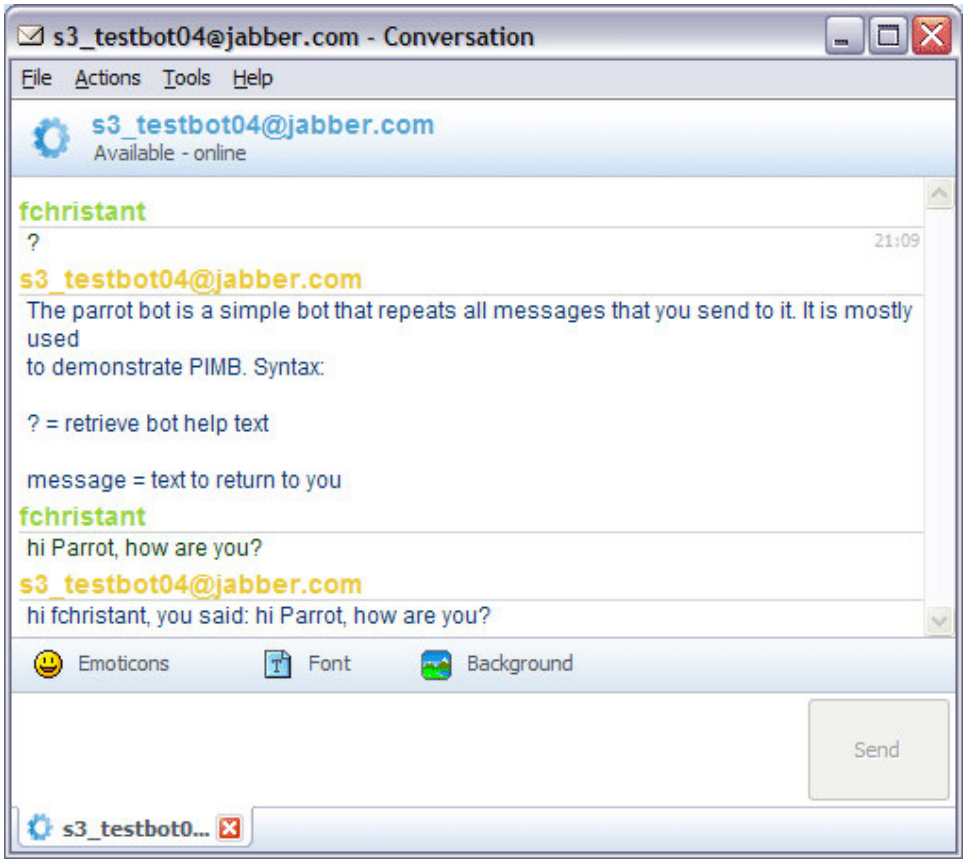


If any message in the log file appears, it means that you have correctly set the write permissions of PIMB. If everything went correctly, the log should indicate that your bot started successfully and that it is waiting for incoming messages. If not, please refer to the troubleshooting section of this manual.

The final test to perform is to interact with the bot using the instant messaging client. Before we can, we need to authorize the human user to use the bot and vice versa. To do so, log on as the bot user in the instant messaging client. Since the human user added the bot earlier, you will have one authorization pending. Accept the authorization. Log on as the human user again and authorize the bot to contact you in a similar way.

Note: You may need to stop and restart the bot from PIMB, since you just logged on as the bot while it was running in a different session.

Finally, we are ready to send a message to the bot. First, enter the bot help command "?". This special command will return the bot usage instructions, which are defined in the bots' ini file. Next, enter any message to the Parrot bot, and it should return it to you. If you got this far, congratulations...PIMB is installed successfully! If not, please consider the troubleshooting section or request support from s3maphor3. Below is a chat transcript from the PSI client with the Parrot bot:



Note: The manual authorization step is likely to disappear in a future version of PIMB.

3. Using PIMB

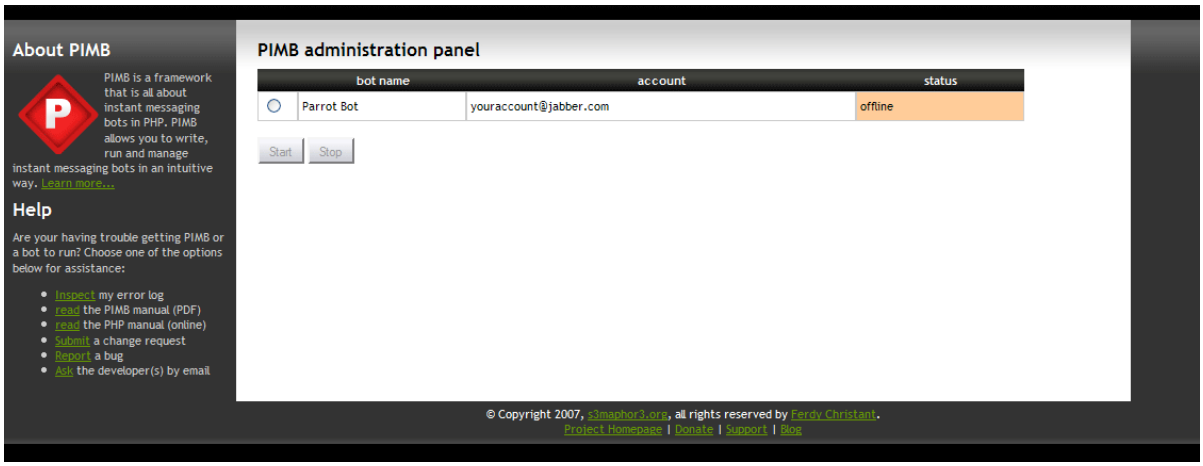
This chapter explains how to use PIMB. PIMB offers functionality to bot administrators to manage the bots hosted by PIMB. PIMB end users typically interact with the bots you have developed using their favourite instant messaging software, while PIMB administrators use the PIMB administration panel to control these bots. The administration panel is simple to use, it offers the following features:

- Viewing bot details
- Starting and stopping bots
- Inspecting bots

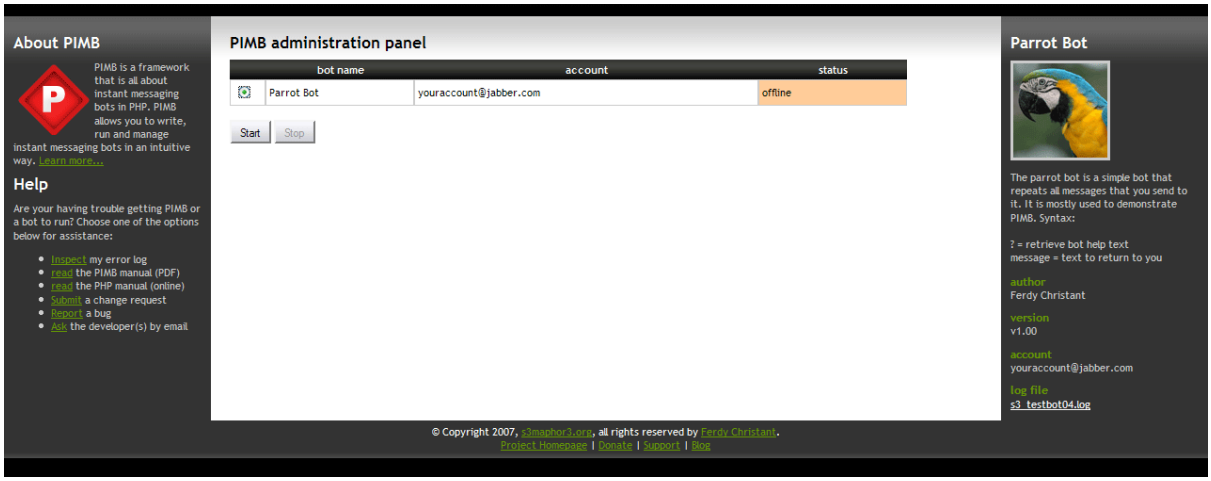
Let's have a look at this functionality in detail.

3.1 Viewing bot details

Open up the PIMB administration panel to have a look at the deployed bots for your PIMB installation. To do this, open the URL to the root of the PIMB directory on your web server, i.e. example.com/PIMB. The result will be similar to this:



In the example above, we see only one bot deployed. It is associated with a certain instant messaging account address, and it is marked offline. To view more details for the bot, simply select the radio button in front of it:



See screenshot. The right panel reveals more details about the bot:

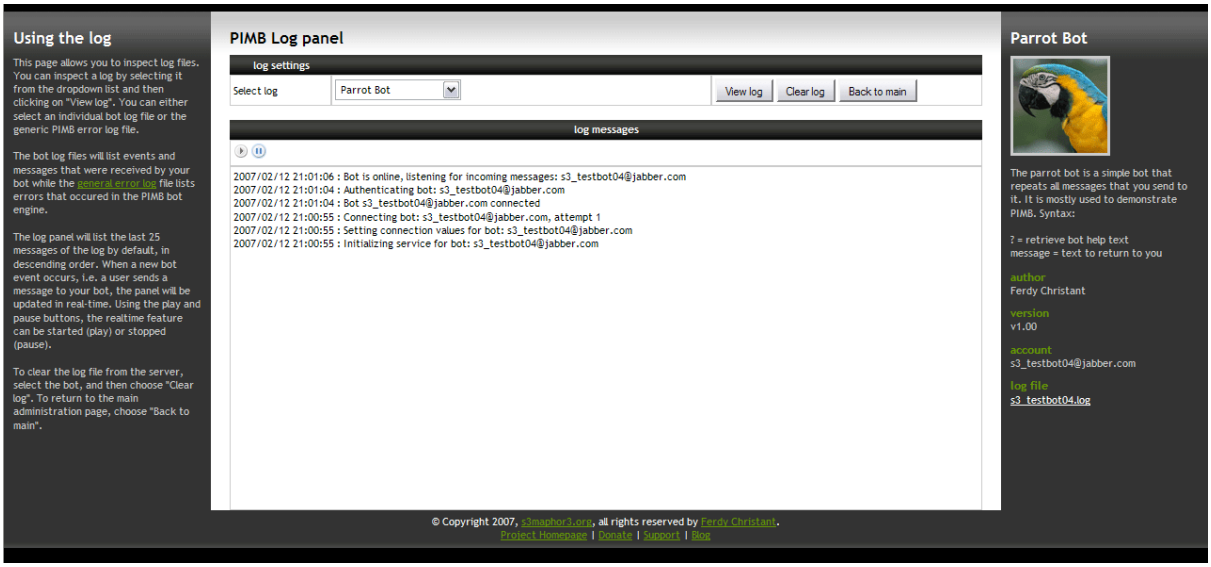
- Avatar. An image of the bot
- Description. The description explains the purpose and the usage of the bot
- Author and version information
- The account associated with the bot
- Link to the log file of the bot

3.2 Starting and stopping bots

To start or stop a bot, simply select the radio button in front of it, and click the start, or stop button. Allow the bot some time to change its state. If any error occurs during the startup of the bot, the status column will indicate this. There is an alternative way to stop a bot. By sending it the message die <bot_password> the bot will be shutdown. The password parameter is the same password that is defined in the bots' ini file.

3.3 Inspecting bots

As mentioned earlier, the bot details panel on the right side of the screen holds a link to the bot's log file. Click on it to open up the log viewer:



The left panel of the log viewer explains in detail how to use the log viewer. The selector in the log viewer allows you to choose any log file of any bot, but also the general PIMB error log. After making your selection, click the "View log" button to open the log file. If you clicked on a bot log link, you do not have to make a selection, as it is automatically set to the bot you have chosen.

The clear log option allows you to delete the log file of the selected bot, but you need to do this with care. Obviously, you will lose the history of the bot. Furthermore, it is best to first stop the bot before deleting the log file of it.

The "Back to main" button brings you back to the main administration panel that lists the bots.

On to the more interesting part of the log viewer: the log message panel. This panel shows the last 25 logged messages of your bot. The value of 25 can be increased or decreased in the "pimb.conf.php" file of your setup. In essence, the log viewer is not suitable for the analysis of massive log files, for that purpose it is best to directly open the log file from the "log" directory of your PIMB installation. Instead, the log viewer is perfect for inspecting recent activity, in real-time. The log panel will be updated in real-time as soon as a new bot event (i.e. a user send a message to your bot) occurs, you do not need to refresh the page.

By default, the real-time behaviour of the log viewer is started. If you want to pause it, you can use the pause icon on top of the panel. To start it again, click the start icon.

Note: Per bot it is optional to enable logging. It is wise to leave it enabled, but not all bots may have a log file.

4. Creating bots

Now that we have learned how to install the software and use the administration panel, it is time for most fun and powerful part of PIMB: creating your own bots. This is really where PIMB is all about: it delivers you the context to create your own bots easily. You can fully focus on your bots' functionality.

If you have no idea how this would be useful, allow me to inspire you. There are millions of users worldwide familiar with instant messaging software to chat with their friends, family and co-workers. Now consider how easy it would be for those users to interact with your bot. All they would need is the bot address to add it to their contact list. Next, they can interact with your bot, enjoying the service it delivers. The simple text based interface is especially suitable for users of mobile devices.

So what functionality will your bot deliver? This is entirely up to you. PIMB gives you the framework, the rest is up to you. In essence, a bot is a PHP script. Anything you can do in PHP, your bot can do. Some examples of possible bots:

- A bot that publishes messages you send to it on a blog
- A survey that interviews users
- A calculator bot that returns the answer to math statements
- A weather bot, returning the weather forecast based on the city you feed it
- A traffic bot, returning a traffic report based on the city you feed it
- A navigator bot, returning driving instructions based on a from and to address
- A AI bot, mimicking a human user through artificial intelligence
- A quote bot, sending you a random quote each time you send it something
- A RPG bot, offering you a classic D&D text-based RPG game ;)

And so on, and so forth. Let's get started.

4.1 Bot basics

Before we go into the coding, we need one step to prepare. Every bot is associated with an instant messaging account. Therefore, you will need to register an instant messaging account. Refer to paragraph 2.5 for instructions on how to register such an account. For now, we will assume that you have registered "example@jabber.com".

As mentioned before, bots are simply PHP scripts that adhere to certain rules enforced by PIMB. In essence, every bot exists of 3 files:

- Botname.ini.php
- Botname.bot.php
- Bot.gif

Typically, each bot is deployed in a separate subdirectory beneath the "bots" directory. So, to create a new bot, you would create a new subdirectory in the bots dir, and then place the three files in it. Note that some bots include other files (for example when your custom bot references "external" PHP scripts), so there may be more than 3 files involved, but for now let's have a look at the files in detail.

4.2 Botname.ini.php

This file contains the settings of your bot. You need to stick to the file naming conventions for bots setting files. If you name your bot "example", the ini file must be named "example.ini.php". The ini file is a PHP file that contains an array of settings for the bot. The following table list the settings:

Setting name	Description
server	The host name of instant messaging account associated with your bot. For example, for the account example@jabber.com, "jabber.com" is the server value. Another possible value would be "gmail.com", for Google Talk bots
bot_name	Short descriptive name of the bot. This value is used in the bot details section of the administration panel
bot_user	The user name of the instant messaging account associated with your bot. For example, for the account example@jabber.com, "example" is the bot user value
bot_password	The password of the instant messaging account associated with your bot
class_name	The name of the class that implements your bot. This class is located in the other file,

Setting name	Description
	example.bot.php. The value is case-sensitive!
resource	How your bot users will perceive the bot account. Leave this value to "bot"
port	This is the port that is used to communicate from your bot script to the server hosting the bot account. In most cases, 5222 is the correct value. There are also Jabber servers with alternative ports, such as jabber80.com.
logging	This setting enables or disables logging. When enabled, your bot will log event, such as the startup of the bot, but also any message traffic between the bot and the users.
log_file	When logging is enabled, this is the log file where the log messages of your bot will be written to. Do not include a path, only the file name. Make sure the file ends with .log.
author	The author of the bot, only used for display in the bot details of the administration panel
version	The version of the bot, only used for display in the bot details of the administration panel
image	The file name of your bot avatar image. Do not include a file path. The avatar will be displayed in the bot details section of the administration panel
description	Text describing your bot's purpose and usage. This will be displayed in the bot details on the administration panel. You can include any HTML.

For an example, you can open up the ini file of the Parrot bot that comes with PIMB. It is located at "bots/parrot/parrot.ini.php".

4.3 Botname.bot.php

This file contains the implementation of your bot. You need to stick to the file naming conventions for bot implementation files, otherwise the administration panel will not pick up your bot as a valid entry. The way to develop a bot implementation file is best explained by going through the example Parrot bot file "bots/parrot/parrot.bot.php":

```
<?php
```

Open a PHP script

```
/*********************************************************************************

        *.bot.php
        This script contains the implementation of your bot
        @author Ferdy Christant <www.ferdychristant.com>
        @copyright Copyright &copy; 2007, Ferdy Christant


/*********************************************************************************

            ===== YOUR BOT IMPLEMENTATION =====

*********************************************************************************/
```

Comments describing the bot. You can change this to your own need.

```
// load interface definition
require_once("inc/class.service.php");
```

This loads the bot API definition. All PIMB bots must comply with this API.

```
class Parrot implements Service {
```

Here we declare our bot implementation class. The class name "Parrot" must match with the class name specified in the ini file we saw earlier. All bot implementation classes are required to implement "Service". Service is the interface that contains the bot API

```
        public function Start() {
                // initialization code. is ran when the bot is started
        }
```

Start is a method you are required to declare. The method is ran when your bot is started. This makes it an ideal place for initialization, for example the opening of a file, or setting up a database connection. If you do not require any initialization code, you can simply leave it empty.

```
	public function Stop() {
		// cleanup code. is ran when the bot is killed
	}
```

Stop is a method you are required to declare. The method is ran when your bot is killed. This makes it an ideal place for cleaning up resources used by the bot, such as open files or database connections. If you do not require any termination code, you can simply leave it empty.

```
	public function Help() {
		// help text to return when users send the help command (?) to this bot
		return "The parrot bot is a simple bot that repeats all messages that you send to it. It is mostly
used
to demonstrate PIMB. Syntax:\n
? = retrieve bot help text\n
message = text to return to you";
	}
```

Help is a method you are required to declare and implement. It is automatically called by the PIMB engine when users send the help command "?" to your bot. The idea of this method is that you return help text to the user that explains them how to use the bot. The return value of the method will be send to the user. In the example above, a multi-line help text is returned to the user that explains them how to use the parrot bot.

```
	public function Call($arg, $from) {
		// logic that is ran when users send a message to your bot.
		// $arg contains the message that was sent to this bot
		// $from is the sender ID
		// anything you return in this function will be send back to $from
		try {
			// extract recipient name from IM address
			$recipient = substr($from, 0, strpos($from, "@"));
			return "hi $recipient, you said: " . $arg;
		} catch (Exception $e) {
			// error occured
			return "error: invalid statement";
		}
	}
```

Call is a method you are required to declare and implement. This is the heart of your bot. Any message send to the bot will be handled by this method, except for the help command. The parameter "$arg" contains the message that was sent to your bot, while "$from" contains the user name of the user sending the message. Anything you return from this message will be the response to the user. Typically, the flow of the call method for a bot looks like this:

- Check/parse the incoming parameter or command that was sent by the user to this bot
- Calculate the response. This is the heart of your bot, the added value your bot delivers
- Return the response to the user

In the example above, the user name is parsed and used to calculate a simple response. Note that you should use an error handling construction as indicated above, otherwise errors will be thrown to the PIMB engine, which may result in the bot being shutdown.

```
}
```

End of bot implementation class.

```
?>
```

End of bot implementation file.

4.4 Bot.gif

The third file is the bot image file. The image file name is not required to be "bot.gif", you can specify this in the ini file of your bot. For the proper display of the image in the bot details section of the administration panel, both the width and height of the image need to be 100 pixels.

4.5 Putting it together

Easy enough, developing a bot is a matter of creating 3 files. With basic PHP scripting skills, you can create your own bots right away. Once you have developed your bot, simply refresh the administration panel. It will now list your new bot. Start the bot, and wait for it to come online. When it is online, you can use your instant messaging client to interact with it. If anything went wrong, please consider the troubleshooting chapter. Do not forget to authorize the bot if you have not done so.

5. Troubleshooting PIMB

While PIMB tries to make your bot experience as easy as possible, things can still go wrong. Below are some common scenarios that may assist you in finding a solution to your problem.

Something went wrong, what do I do now?

First, go through this manual in detail to see if it contains the answer. Try to get as much logging information from PIMB as possible. If that still does not work, consider requesting support, as explained in chapter 1.

My bot does not start, now what?

First, check the log of the bot. If it mentions a connection problem, it means that PIMB is unable to connect to the server that hosts the bot instant messaging account. Check your server's connection to the internet. Make sure that your firewall allows traffic from the port mentioned in your bot's ini file. Also make sure that the server hosting your bot account is up.

Another cause to this problem may be in your system specifications. Note that PIMB requires a minimum of PHP5 with safe mode disabled. It is also possible that you are running PIMB on a system for which PIMB is not tested.

Finally, it is possible that you made an error in the settings or implementation of your bot. Step 1 is to get rid of compilation errors, which is easy in IDEs like Zend studio. Step 2 is to check how your code behaves in run-time. You can do this by adding extra log statements in your code. An alternative is to run your bot from a command line. This can be accomplished by navigating to your PIMB directory on the web server and issuing the following command:

```
php startbot.php bots/botname/botname.bot.php
```

Next, the console will report any errors that occur, including echo statements that you may have inserted for the purpose of debugging.

My bot starts, but shutdowns automatically

A bot is a PHP script that runs in a continuous loop, waiting for user messages. Some hosts do not allow a PHP script to run longer than a specified time. PIMB set the maximum execution time of bot scripts to 0, meaning unlimited. This is only possible when safe mode is disabled.

If the problem persists, or if you run PIMB in an environment where you cannot disable safe mode, the alternative is to start each bot from the command line. Navigate to the PIMB directory on your web server and issue the following command:

```
php startbot.php bots/botname/botname.bot.php &
```

This will bypass the restricted time limit.

My bot does not respond to my instant messaging client

PIMB is currently only developed for Jabber clients, like PSI, and some Jabber-compatible clients, such as Google Talk. Networks that are not compatible with Jabber, such as MSDN, will not work with your bot. This is a restriction of the network, not of PIMB.

My bot is marked as online in the administration panel, but in reality it is offline

This situation can occur when a bot is shutdown without PIMB being able to detect it, for example, when the server hosting the bot account is temporarily unavailable. The easiest way to correct the situation is to stop the bot from the administration panel, and restarting it. An alternative is to clear the lock file in the "locks" directory. These lock files indicate that PIMB considers a bot to be online.

S3MAPHOR3.ORG