

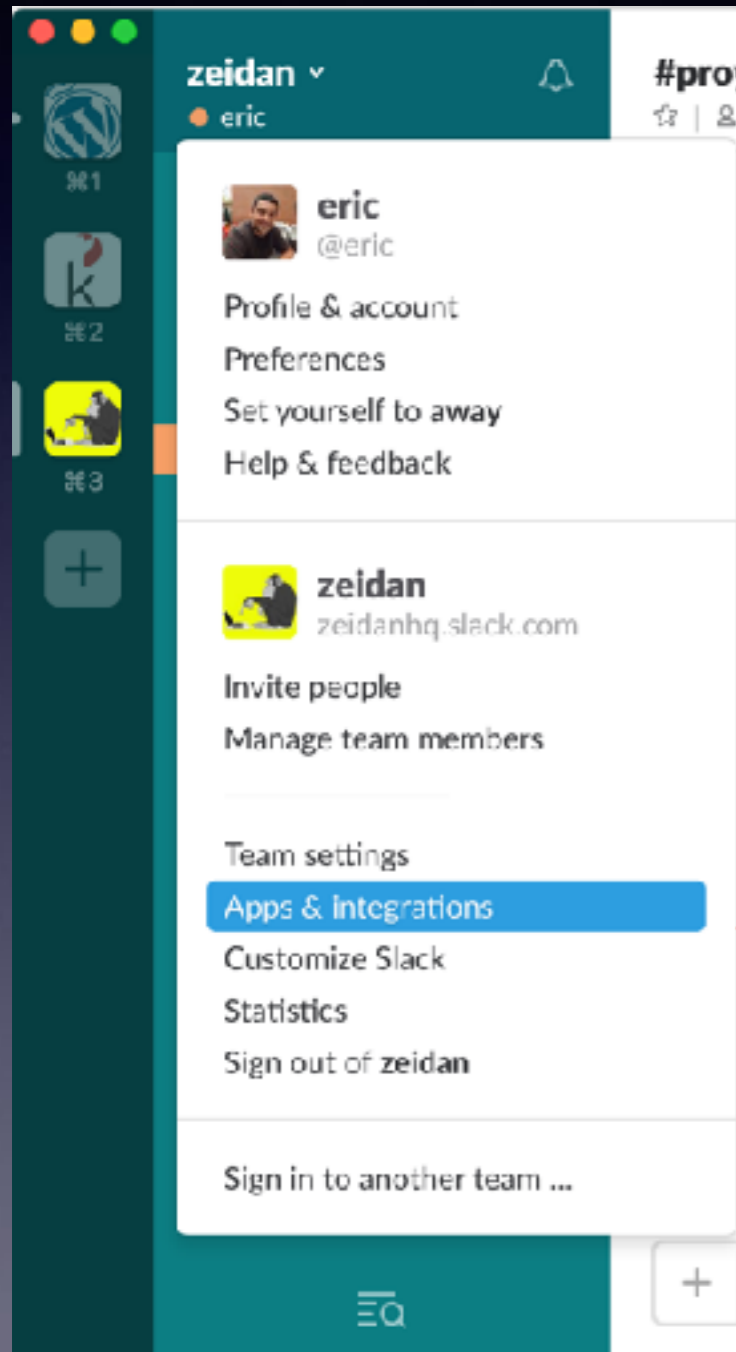


Slack, WordPress y un poco más...

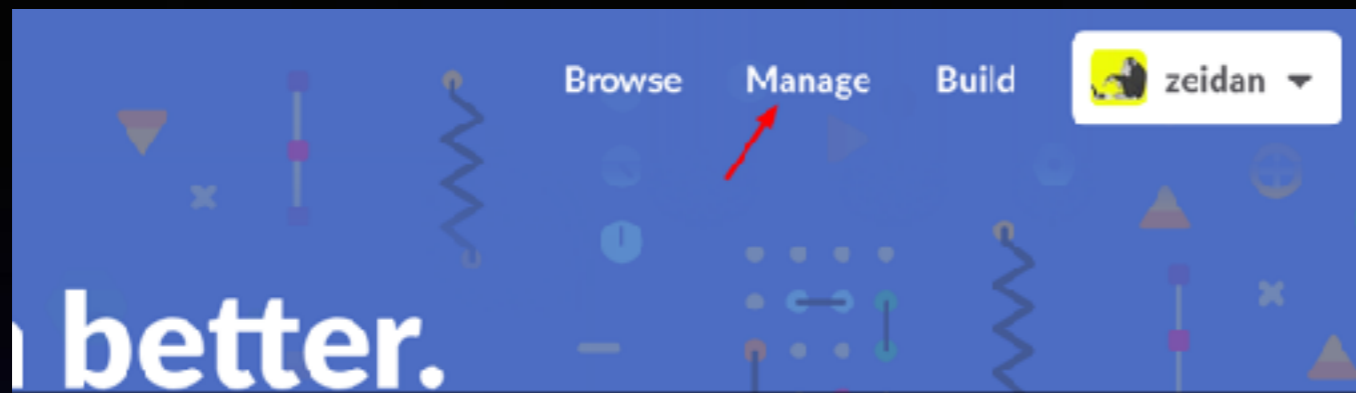
Eric Zeidan

<https://twitter.com/ericjanzei>

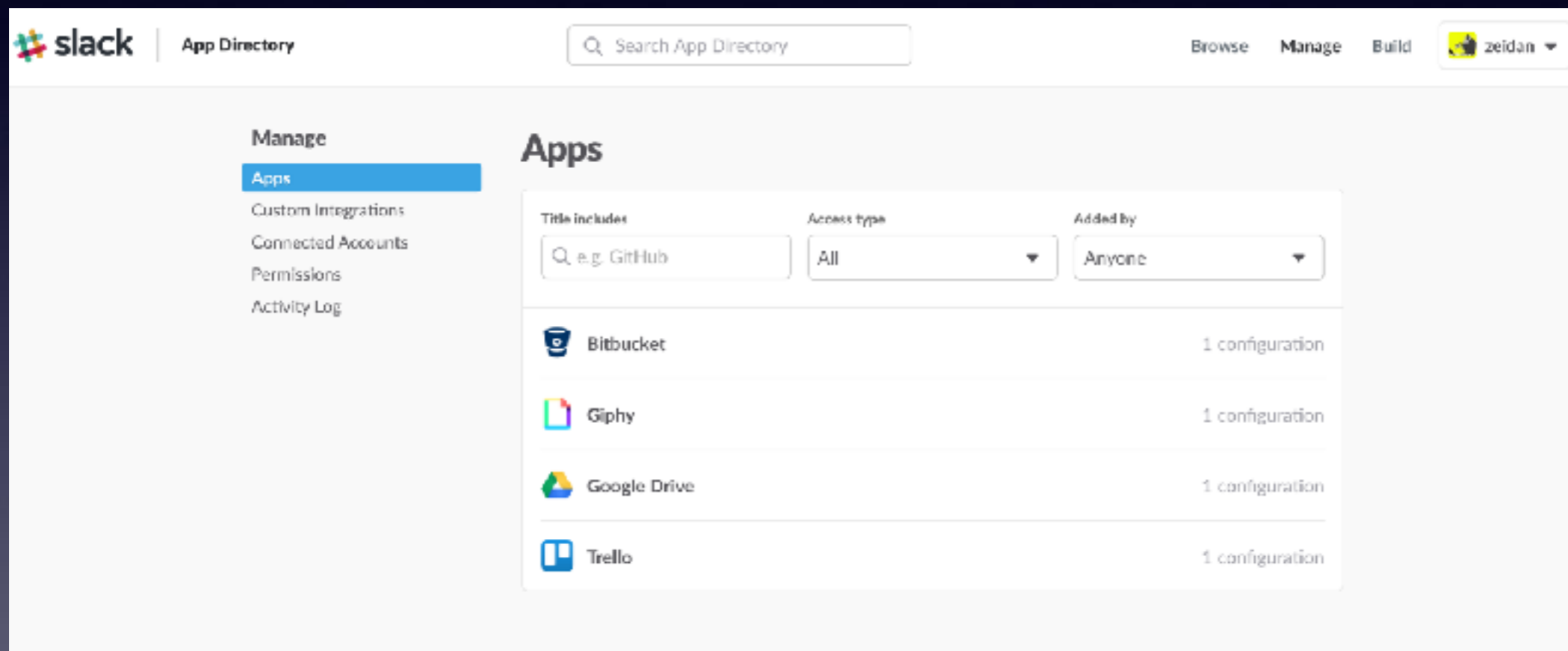
Como integramos WordPress u otra aplicación a Slack?



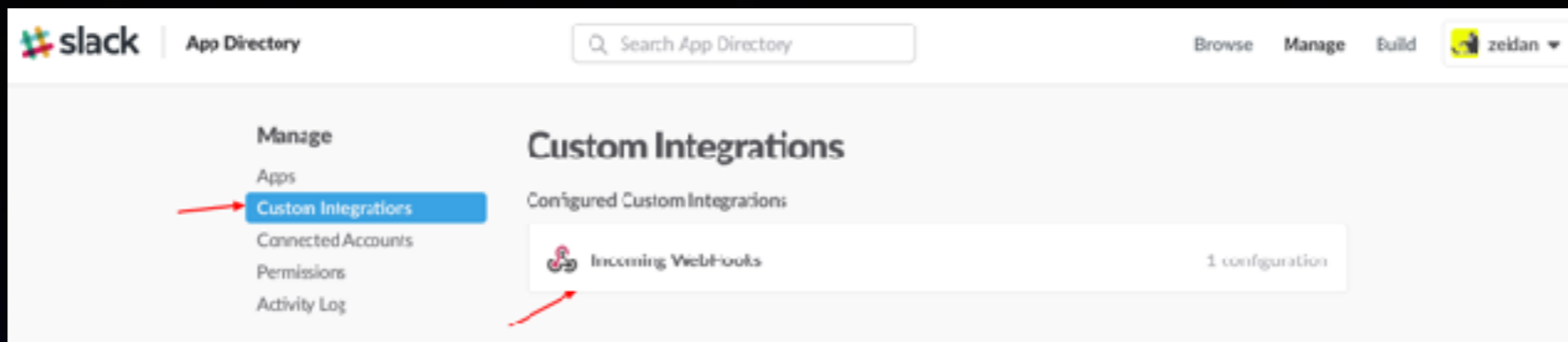
- Lo primero que debemos hacer es abrir nuestro Slack, y haciendo click sobre nuestro Usuario nos aparecerá la opción de Apps & Integrations, hacemos click sobre ella y nos abrirá la página de Slack Apps & Integrations.



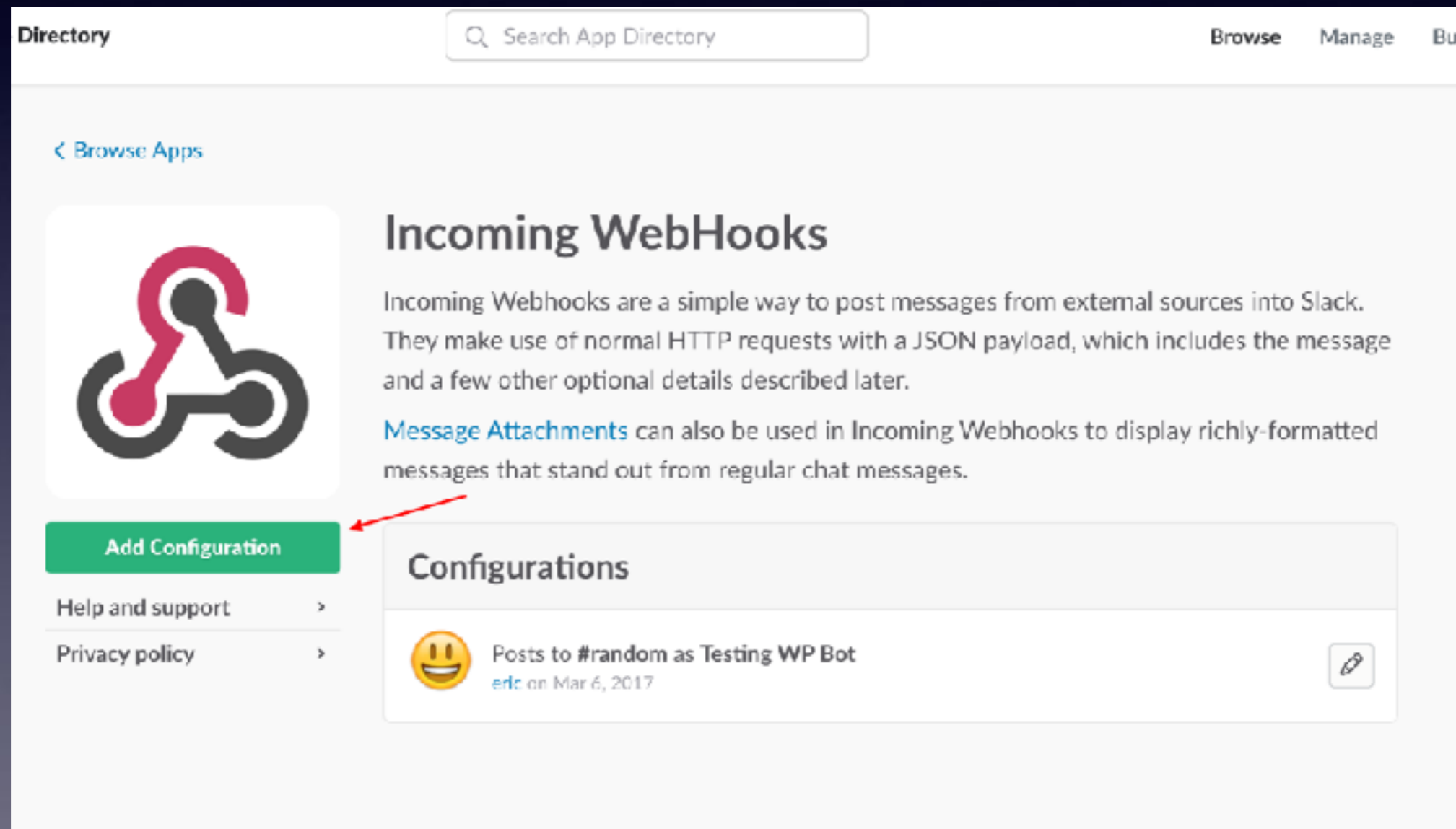
Al ir a la página haremos clic sobre Manage



Nos abrirá las opciones de manejo, entre ellas la de Apps (Aplicaciones) y la que nos interesa que será la de Custom Integrations (Integraciones Personalizadas)




Vemos nuestros Incoming WebHooks



Directory Browse Manage B

Incoming Webhooks are a simple way to post messages from external sources into Slack. They make use of normal HTTP requests with a JSON payload, which includes the message and a few other optional details described later.

[Message Attachments](#) can also be used in Incoming Webhooks to display richly-formatted messages that stand out from regular chat messages.

 **New to Slack integrations?**
Check out our [Getting Started](#) guide to familiarize yourself with the most common types of integrations, and tips to keep in mind while building your own. You can also [register as a developer](#) to let us know what you're working on, and to receive future updates to our APIs.

Post to Channel

Start by choosing a channel where your Incoming Webhook will post messages to.

[or create a new channel](#)

[Add Incoming WebHooks integration](#)

By creating an incoming webhook, you agree to the [Slack API Terms of Service](#).

Veamos en vivo como configuramos el Webhook

Ahora veamos como haremos la magia desde WordPress

Para ello usaremos la siguiente función donde pasaremos vía wp_remote_get (función de cUrl de WP) y pasaremos los parámetros necesarios por Slack para mostrar nuestro mensaje.

```
//Función para enviar mensajes por Slack desde WordPress
function slack_send($message, $room = "engineering", $icon = ":robot_face:") {
    $room = ($room) ? $room : "engineering";
    $data = "payload=" . json_encode(array(
        // "channel" => "#{$room}",
        "text" => $message,
        "icon_emoji" => $icon
    ));

    $url = ""; // tu URL de Slack
    $args = array(
        'method' => 'POST',
        'timeout' => 45,
        'redirection' => 5,
        'httpversion' => '1.0',
        'body' => $data,
        'cookies' => array()
    );

    $result = wp_remote_post($url, $args);

    return $result; }

```

Como sería esta función en cUrl?

```
function slack($message, $room = "it", $icon = ":longbox:") {
    $room = ($room) ? $room : "engineering";
    $data = "payload=" . json_encode(array(
//      "channel"      => "#{$room}",
        "text"         => $message,
        "icon_emoji"   => $icon
    ));

    $ch = curl_init("https://hooks.slack.com/services/T388TFUE8/B3WDS2X7F/XcJJkw
zEPIESs8Qwq79J52xu");
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "POST");
    curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    $result = curl_exec($ch);
    curl_close($ch);

    return $result;
}
```


Para modificar una función de WordPress, tomamos ventaja del set de funciones y realizamos un plugin donde haremos un sobre escritura de la función.

```
if(!function_exists('wp_new_user_notification')) {
function wp_new_user_notification( $user_id, $plaintext_pass = '' ) {
    $user = new WP_User($user_id);

    $user_login = stripslashes($user->user_login);
    $user_email = stripslashes($user->user_email);

    $message = sprintf(__('New user registration on your blog %s:'), get_option('blogname')) . "\r\n\r\n";
    $message .= sprintf(__('Username: %s'), $user_login) . "\r\n\r\n";
    $message .= sprintf(__('E-mail: %s'), $user_email) . "\r\n";

    //@wp_mail(get_option('admin_email'), sprintf(__('[%s] New User is Register Eric'), get_option('blogname')), $message);
    plug_slack_send(sprintf(__('[%s] New User is Register'), get_option('blogname')));

    if ( empty($plaintext_pass) )
        return;

    $message = __('Hi there,') . "\r\n\r\n";
    $message .= sprintf(__('Welcome to %s! Here's how to log in eric:'), get_option('blogname')) . "\r\n\r\n";
    $message .= wp_login_url() . "\r\n";
    $message .= sprintf(__('Username: %s'), $user_login) . "\r\n";
    $message .= sprintf(__('Password: %s'), $plaintext_pass) . "\r\n\r\n";
    $message .= sprintf(__('If you have any problems, please contact me at %s.'), get_option('admin_email')) . "\r\n\r\n";
    $message .= __('Adios!');

    //wp_mail($user_email, sprintf(__('[%s] Your username and password eric'), get_option('blogname')), $message);
    plug_slack_send(sprintf(__('[%s] New User is Register'), $message));
}

function plug_slack_send($message, $room = "engineering", $icon = ":robot_face:") {
    $room = ($room) ? $room : "engineering";
    $data = "payload=" . json_encode(array(
        // "channel" => "#{$room}",
        "text" => $message,
        "icon_emoji" => $icon
    ));

    $url = "https://hooks.slack.com/services/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"; // tu URL de Slack
    $args = array(
        'method' => 'POST',
        'timeout' => 45,
        'redirection' => 5,
        'httpversion' => '1.0',
        'body' => $data,
        'cookies' => array()
    );
    $result = wp_remote_post($url, $args);

    return $result;
}
```


Ademas de usarlo para WordPress podemos usar cUrl para enviar cualquier mensaje a Slack desde PHP, inclusive desde lineas de comando en el servidor o como desde Shell Scripts

Veamos un ejemplo de sh desde linux

```
$ curl -X POST -H 'Content-type: application/json' --data '{"text" : "Test trying from server"}'  
https://hooks.slack.com/services/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
```

Este es un ejemplo de sh para medir la carga del servidor

```
#!/bin/bash  
trigger=4.00  
load=`cat /proc/loadavg | awk '{print $1}'`  
response=`echo | awk -v T=$trigger -v L=$load 'BEGIN{if ( L > T){ print "greater"}}`  
if [[ $response = "greater" ]]  
then  
curl -X POST -H 'Content-type: application/json' --data '{"text" : "High load on server"}' https://hooks.slack.com/  
services/T3323288TFUE8/B4dewdew88USGCB/FgLmeqwqwsUko2xF12H5EnaFaZDAm8  
fi
```

Otro ejemplo de sh, backup de base de datos con subida a git y envío de mensaje por Slack

```
#!/bin/bash
DIR=$(cd -P -- "$(dirname -- "$0")" && pwd -P)
DB_USER=$1
DB_PASS=$2
DB=$3
FILENAME="$3.sql"
MESSAGE=`date +%e-%m-%y`
cd $(dirname $(readlink -f $0))
`mysqldump -u $DB_USER -p$DB_PASS $DB > $DIR/$FILENAME`
RESULT=$(git add *.sql && git commit -a -m "$MESSAGE" && git pull --no-edit && git push)
echo $RESULT
curl -X POST -H 'Content-type: application/json' --data '{"text" : "Database ""${3}"" backed up date ""${MESSAGE}"""}' https://hooks.slack.com/services/T388TwqwqwFUE8/B4C0Mw23qwq3PK4/zcGAcZIUwqwqwewqe3ee1qBUHU8sdyoRSdmzBn
```

Gracias

@ericjanzei

<https://zeidan.info>