**Manual de Configuración de R.A.R.**

**Códigos PHP**

Esta función guarda los logs de las personas que ingresan a la base de datos, los datos que se guardan son:

* Nombre
* IP
* Fecha y hora

require("config.inc.php");

function GetUserIP() {

if (isset($\_SERVER["HTTP\_CLIENT\_IP"]))

{

return $\_SERVER["HTTP\_CLIENT\_IP"];

}

elseif (isset($\_SERVER["HTTP\_X\_FORWARDED\_FOR"]))

{

return $\_SERVER["HTTP\_X\_FORWARDED\_FOR"];

}

elseif (isset($\_SERVER["HTTP\_X\_FORWARDED"]))

{

return $\_SERVER["HTTP\_X\_FORWARDED"];

}

elseif (isset($\_SERVER["HTTP\_FORWARDED\_FOR"]))

{

return $\_SERVER["HTTP\_FORWARDED\_FOR"];

}

elseif (isset($\_SERVER["HTTP\_FORWARDED"]))

{

return $\_SERVER["HTTP\_FORWARDED"];

}

else

{

return $\_SERVER["REMOTE\_ADDR"];

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet, el numero de la lookback que se va a crear y la ip y la máscara de la lookback que se va a crear.

Mediante la función fsockopen se ingresa al router a través telnet y se le mandan los comandos respectivos para la creación de una lookback con su ip y mascara.

function crearLoopBack($ip\_router,$numero\_loopback,$ip\_loopback,$mascara\_loopback){

$port = 23;

$timeout = 10;

$router\_ip = $ip\_router;

$username = "adm";

$password = "adm";

$connection = fsockopen($router\_ip, $port, $errno, $errstr, $timeout);

if(!$connection){

return $ip\_router;

} else {

echo "Connected\n";

fputs($connection, $username."\r\n");

fputs($connection, $password."\r\n");

fputs($connection, "config t\r\n");

fputs($connection, "interface loopback".$numero\_loopback."\r\n");

fputs($connection, "ip address ".$ip\_loopback." ".$mascara\_loopback."\n");

fputs($connection, "exit\r\n");

$j = 0;

while ($j < 16) {

fgets($connection, 20);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

return "Connected\n";

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet, la ip del router al cual queremos hacer ping y la ip origen para hacer un ping extendido.

Mediante la función fsockopen se ingresa al router a través telnet y se le mandan los comandos respectivos para realizar un ping extendido a la ip previamente ingresada.

function hacerPing($ip\_router,$ip\_destino,$ip\_origen){

$resultado="";

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return "Error!";

} else {

fputs($connection, "$username\r\n");

fputs($connection, "$password\r\n");

fputs($connection, "ping ".$ip\_destino.” ”.”source ”.$ip\_origen."\r\n");

fputs($connection, " ");

$j = 0;

while ($j < 7) {

fgets($connection, 128);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

$j=0;

while ($j < 4) {

$resul=fgets($connection, 128);

$resultado=$resultado.$resul."\n";

$j++;

}

return $resultado;

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet, la interface la cual queremos levantar.

Mediante la función fsockopen se ingresa al router a través telnet y se le mandan los comandos respectivos para levantar una interface mediante el comando no shutdown.

function noShutdown($ip\_router,$interface){

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return $ip\_router;

exit();

} else {

fputs($connection, $username."\r\n");

fputs($connection, $password."\r\n");

fputs($connection, "config t\r\n");

fputs($connection, "interface ".$interface."\r\n");

fputs($connection, "no shutdown \r\n ");

fputs($connection, "exit\r\n");

$j = 0;

while ($j < 16) {

fgets($connection, 20);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

return "Connected\n";

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet, la interface la cual queremos dar de baja.

Mediante la función fsockopen se ingresa al router a través telnet y se le mandan los comandos respectivos para dar de baja una interface mediante el comando shutdown.

function shutdown($ip\_router,$interface){

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return $ip\_router;

exit();

} else {

echo "Connected\n";

fputs($connection, $username."\r\n");

fputs($connection, $password."\r\n");

fputs($connection, "config t\r\n");

fputs($connection, "interface ".$interface."\r\n");

fputs($connection, "shutdown \r\n ");

fputs($connection, "exit\r\n");

$j = 0;

while ($j < 16) {

fgets($connection, 20);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

return "Connected\n";

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet.

Mediante la función fsockopen se ingresa al router a través telnet y se le mandan los comandos para configurar la plantilla básica en el router.

function crearPlantillaBasica($ip\_router){

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return "Error de Conexion";

} else {

fputs($connection, $username."\r\n");

fputs($connection, $password."\r\n");

fputs($connection, "config t\r\n");

fputs($connection, "username adm privilege 15 secret adm\r\n");

fputs($connection, "username monitoreo privilege 5 secret monitoreo\r\n");

fputs($connection, "banner motd #ACCESO RESTRINGIDO SOLO A PERSONAL AUTORIZADO#\r\n");

fputs($connection, "no ip domain-lookup\r\n");

fputs($connection, "ip name-server 200.93.195.254\r\n");

fputs($connection, "ip domain-name espol.com\r\n");

fputs($connection, "line console 0\r\n");

fputs($connection, "login local\r\n");

fputs($connection, "exec-timeout 3 3\r\n");

fputs($connection, "line vty 0 4\r\n");

fputs($connection, "login local\r\n");

fputs($connection, "exec-timeout 3 3\r\n");

fputs($connection, "exit\r\n");

fputs($connection, "end\r\n");

fputs($connection, "wr\r\n");

$j = 0;

while ($j < 16) {

fgets($connection, 150);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

return "Connected\n";

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet, la ip y la máscara de la red a la que queremos llegar y el siguiente salto.

Mediante la función fsockopen se ingresa al router a través telnet y se le mandan los comandos respectivos para dar de baja una interface mediante el comando shutdown.

function crearRutaEstatica($ip\_router,$ip\_destino,$mascara,$next\_hop){

$port = 23;

$timeout = 10;

$router\_ip = $ip\_router;

$username = "adm";

$password = "adm";

$connection = fsockopen($router\_ip, $port, $errno, $errstr, $timeout);

if(!$connection){

return $ip\_router;

exit();

} else {

echo "Connected\n";

fputs($connection, $username."\r\n");

fputs($connection, $password."\r\n");

fputs($connection, "config t\r\n");

fputs($connection, "ip route ".$ip\_destino." ".$mascara." ".$next\_hop."\r\n");;

$j = 0;

while ($j < 16) {

fgets($connection, 30);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

return "Connected\n";

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet.

Mediante la función fsockopen se ingresa al router a través telnet, se le manda el comando de show running-config, con un while va tomado cada línea del show, lo concatena en un variable string para al final mandarla a la aplicación y que se muestre en pantalla.

function showRun($ip\_router){

$resultado="";

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return "Error";

} else {

fputs($connection, "$username\r\n");

fputs($connection, "$password\r\n");

fputs($connection, "show running-config\r\n");

fputs($connection, " ");

$j = 0;

while ($j < 16) {

fgets($connection, 128);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

while (!feof($connection)) {

$resul=fgets($connection, 128);

$info = stream\_get\_meta\_data($connection);

if (ereg('#', $resul) ){

break;

}

elseif (ereg('--More--', $resul) ){

fputs ($connection, " ");

}

else{

$resultado=$resultado.$resul."\n";

}

if ($info['timed\_out']) {

$timeoutCount++; // We want to count, how many times repeating.

}

if ($timeoutCount >2){ // If repeating more than 2 times,

break; // the connection terminating..

}

}

return $resultado;

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet.

Mediante la función fsockopen se ingresa al router a través telnet, se le manda el comando de show ip interface brief para mostrar un resumen de las interfaces del router, con un while va tomado cada línea del show, lo concatena en un variable string para al final mandarla a la aplicación y que se muestre en pantalla.

function showInterfaces($ip\_router){

$resultado="";

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return "Error";

} else {

fputs($connection, "$username\r\n");

fputs($connection, "$password\r\n");

fputs($connection, "show ip interface brief\r\n");

fputs($connection, " ");

$j = 0;

while ($j < 8) {

$resul=fgets($connection, 50);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

while (!feof($connection)) {

$resul=fgets($connection, 128);

$info = stream\_get\_meta\_data($connection);

if (ereg('#', $resul) ){

break;

}

else{

$resultado=$resultado.$resul."\n";

}

if ($info['timed\_out']) {

$timeoutCount++; // We want to count, how many times repeating.

}

if ($timeoutCount >2){ // If repeating more than 2 times,

break; // the connection terminating..

}

}

return $resultado;

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet.

Mediante la función fsockopen se ingresa al router a través telnet y se le manda el comando de show ip route para mostrar la tabla de enrutamiento del router , con un while va tomado cada línea del show, lo concatena en un variable string para al final mandarla a la aplicación y que se muestre en pantalla.

function showRoute($ip\_router){

$resultado="";

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return "Error";

} else {

fputs($connection, "$username\r\n");

fputs($connection, "$password\r\n");

fputs($connection, "show ip route\r\n");

fputs($connection, " ");

$j = 0;

while ($j < 8) {

$resul=fgets($connection, 50);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

while (!feof($connection)) {

$resul=fgets($connection, 128);

$info = stream\_get\_meta\_data($connection);

if (ereg('#', $resul) ){

break;

}

else{

$resultado=$resultado.$resul."\n";

}

if ($info['timed\_out']) {

$timeoutCount++; // We want to count, how many times repeating.

}

if ($timeoutCount >2){ // If repeating more than 2 times,

break; // the connection terminating..

}

}

return $resultado;

}

}

Esta función recibe como parámetros la ip del router al que vamos a configurar mediante telnet.

Mediante la función fsockopen se ingresa al router a través telnet y se le manda el comando de show process cpu history para mostrar un resumen del estado del CPU del router y con un while va tomado cada línea del show, lo concatena en un variable string para al final mandarla a la aplicación y que se muestre en pantalla.

function showProcess($ip\_router){

$resultado="";

$port = 23;

$timeout = 10;

$username = "adm";

$password = "adm";

$connection = fsockopen($ip\_router, $port, $errno, $errstr, $timeout);

if(!$connection){

return "Error";

} else {

fputs($connection, "$username\r\n");

fputs($connection, "$password\r\n");

fputs($connection, "show process cpu history\r\n");

$j = 0;

while ($j < 8) {

$resul=fgets($connection, 50);

$j++;

}

stream\_set\_timeout($connection, 2);

$timeoutCount = 0;

while (!feof($connection)) {

$resul=fgets($connection, 128);

$info = stream\_get\_meta\_data($connection);

if (ereg('#', $resul) and !ereg(' # ', $resul)){

break;

}

elseif (ereg('--More--', $resul) ){

fputs ($connection, " ");

}

else{

$resultado=$resultado.$resul."\n";

}

if ($info['timed\_out']) {

$timeoutCount++; // We want to count, how many times repeating.

}

if ($timeoutCount >2){ // If repeating more than 2 times,

break; // the connection terminating..

}

}

return $resultado;

}

}

Este código ejecuta a las funciones previamente creadas dependiente del parámetro “función” y varios “if” el cual nos dice cual función llamar. Aquí se reciben los parámetros que serán enviados a las funciones mediante mensajes JSON a través del comando $POST.

$ip = GetUserIP();

if (!empty($\_POST)) {

if ($\_POST['funcion']==1){

//obtenemos los usuarios respecto a la usuario que llega por parametro

$query = "

SELECT

name,

pass

FROM users

WHERE

name = :username

";

$query\_params = array(

':username' => $\_POST['username']

);

try {

$stmt = $db->prepare($query);

$result = $stmt->execute($query\_params);

}

catch (PDOException $ex) {

//para testear pueden utilizar lo de abajo

//die("la consulta murio " . $ex->getMessage());

$response["success"] = 0;

$response["message"] = "Problema con la base de datos, vuelve a intetarlo";

die(json\_encode($response));

}

//la variable a continuación nos permitirará determinar

//si es o no la información correcta

//la inicializamos en "false"

$validated\_info = false;

//vamos a buscar a todas las filas

$row = $stmt->fetch();

if ($row)

{

//encaso que no lo este, solo comparamos como acontinuación

if (md5($\_POST['password']) == $row['pass'])

{

$login\_ok = true;

}

}

if ($login\_ok)

{

$query = "

INSERT INTO

log

(name,IP,logTime)

VALUES

(:username,:ip, NOW())

";

$query\_params = array(

':username' => $\_POST['username'],

':ip'=>$ip

);

try

{

$stmt = $db->prepare($query);

$result = $stmt->execute($query\_params);

}

catch (PDOException $ex)

{

$response["success"] = 0;

$response["message"] = "Problema con la base de datos, vuelve a intentarlo";

die(json\_encode($response));

}

$response["success"] = 1;

$response["message"] = "Login correcto!";

die(json\_encode($response));

}

else

{

$response["success"] = 0;

$response["message"] = "Login INCORRECTO";

die(json\_encode($response));

}

}

elseif ($\_POST['funcion']==2){ //loopback

$ip\_router=$\_POST['router'];

$numero\_loopback=$\_POST['numeroLoopback'];

$ip\_loopback=$\_POST['ip'];

$mascara\_loopback=$\_POST['mascara'];

$response["success"] = 1;

$response["message"] = crearLoopBack($ip\_router,$numero\_loopback,$ip\_loopback,$mascara\_loopback);

die(json\_encode($response));

}

elseif ($\_POST['funcion']==3){//ping

$ip\_router= $\_POST['router'];

$ip\_destino=$\_POST['ipDestino'];

$ip\_origen=$\_POST['ipOrigen'];

$response["success"] = 1;

$response["message"] ="Exito";

$resul = hacerPing($ip\_router,$ip\_destino,$ip\_origen);

$response["text"]=$resul;

die(json\_encode($response));

}

elseif ($\_POST['funcion']==4){ //no shutdown

$ip\_router=$\_POST['router'];

$interface=$\_POST['interface'];

$response["success"] = 1;

$response["message"] = noShutdown($ip\_router, $interface);

die(json\_encode($response));

}

elseif ($\_POST['funcion']==5){ // shutdown

$ip\_router=$\_POST['router'];

$interface=$\_POST['interface'];

$response["success"] = 1;

$response["message"] = shutdown($ip\_router, $interface);

die(json\_encode($response));

}

elseif ($\_POST['funcion']==6){ //plantilla basica

$ip\_router=$\_POST['router'];

$response["success"] = 1;

$response["message"] = crearPlantillaBasica($ip\_router);

die(json\_encode($response));

}

elseif ($\_POST['funcion']==8){ //crear ruta estatica

$ip\_router=$\_POST['router'];

$ip\_destino=$\_POST['ipDestino'];

$mascara=$\_POST['mascara'];

$next\_hop=$\_POST['nextHop'];

$response["success"] = 1;

$response["message"] = crearRutaEstatica($ip\_router,$ip\_destino,$mascara,$next\_hop);

die(json\_encode($response));

}

elseif ($\_POST['funcion']==9){

$ip\_router=$\_POST['router'];

$response["success"] = 1;

$response["message"] ="Exito";

$resul = showRun($ip\_router);

$response["text"]=$resul;

die(json\_encode($response));

}

elseif ($\_POST['funcion']==10){

$ip\_router=$\_POST['router'];

$response["success"] = 1;

$resul = showInterfaces($ip\_router);

$response["text"]=$resul;

die(json\_encode($response));

}

elseif ($\_POST['funcion']==11){

$ip\_router=$\_POST['router'];

$response["success"] = 1;

$resul = showProcess($ip\_router);

$response["text"]=$resul;

die(json\_encode($response));

}

elseif ($\_POST['funcion']==12){

$ip\_router=$\_POST['router'];

$response["success"] = 1;

$resul = showRoute($ip\_router);

$response["text"]=$resul;

die(json\_encode($response));

}

}

else

{

?>

<h1>Login</h1>

<form action="index.php" method="post">

Username:<br />

<input type="text" name="username" placeholder="username" />

<br /><br />

Password:<br />

<input type="password" name="password" placeholder="password" value="" />

<br /><br />

<input type="submit" value="Login" />

</form>

<a href="register.php">Register</a>

<?php

}