



**UNIVERSIDAD DE GUAYAQUIL**  
**Facultad de Ciencias Matemáticas y Físicas**  
**Carrera de Ingeniería en Sistemas Computacionales**

Configuración de Redes VPN (Redes Privadas Virtuales) y  
Creación de Clientes VPN – MyVPN

**PROYECTO DE GRADO**

**Previo a la obtención del Título de:**

**INGENIERO EN SISTEMAS COMPUTACIONALES**

**Autores:**

Miguel Ángel Contreras Jordán

Eduardo Xavier Guambo Erazo

David Rafael Morante Lombeida

**GUAYAQUIL - ECUADOR**

**Año 2008**

## **AGRADECIMIENTOS**

En primer lugar agradezco a Dios por haberme dado las fuerzas necesarias para seguir con mis estudios y no desmayar, a mis padres por apoyarme incondicionalmente en esta etapa tan importante de mi vida, a mi esposa e hijos que me supieron comprender cuando no estaba con ellos pero todo tiene su recompensa y esta es la mejor, esto va por ellos Miguel Andrés y Juan Ángel, los quiero mucho.

Miguel Ángel Contreras Jordán

## AGRADECIMIENTOS

A Dios por regalarme unos padres maravillosos que me han ayudado en todo momento. A mis hermanos: Fernando y Paola, que creyendo en mí, me llenaron de fuerza. A mi tía la Lcda. Rosario Erazo, mi segunda mamá, le debo tanto también. Al P. Fray Julio Landívar y a la Comunidad de Agustinos, que me proveyeron el mejor consejo en mis momentos de zozobra. A todos los profesores que me inculcaron sus conocimientos en mi etapa universitaria y a tantos buenos amigos y amigas que hice en la misma. A todos ellos, mi sincero y eterno agradecimiento.

*Xavier Guambo*

## **AGRADECIMIENTOS**

A Dios sobre todas las cosas, a mi madre, la Ing. Shirley Lombeida, quien luchó conmigo día a día en este camino de esfuerzo y perseverancia; a toda mi familia que siempre está a mi lado para resolver cualquier problema. A la Ing. Giselly Ycaza M. por esos pequeños detalles que terminan siendo los más importantes, a mis amigos, compañeros y profesores los cuales han contribuido en la culminación de mi carrera.

David Rafael Morante Lombeida.

## **DEDICATORIAS**

A Dios, a mis padres y a todas aquellas personas que estuvieron a mi lado cuando más las necesite.

Miguel Ángel Contreras Jordán

## DEDICATORIAS

Definitivamente este logro va dedicado a Dios, Fuente de Toda Vida y; a mis padres, el Lcdo. Víctor Hugo Guambo y la Ing. Com. Ana María Erazo, quienes soñaron con este momento con tanta ilusión y entusiasmo, incluso más que yo, a ellos les debo todo lo que soy. Amados padres, espero devolverles con la obtención de este título, al menos un poco de toda la alegría y el bienestar que me han procurado desde que vine a este mundo.

*Xavier Guambo*

## **DEDICATORIAS**

A mis abuelos quienes fueron pilares en este logro, a mi novia Mayra Ajón Y., quien fue mi apoyo en todo momento.

David Rafael Morante Lombeida

## TRIBUNAL DE GRADUACIÓN

---

Presidente del Tribunal

---

Primer Vocal

---

Segundo Vocal

---

Secretario

## DECLARACION EXPRESA

“La autoría de la tesis de grado corresponde exclusivamente al suscrito(s), perteneciendo a la Universidad de Guayaquil los derechos que generen la aplicación de la misma”.

(Reglamento de Graduación de la Carrera de Ingeniería en Sistemas Computacionales, Art. 26).

---

Miguel Angel Contreras Jordán  
mike\_contreras\_jordan@hotmail.com

---

Eduardo Xavier Guambo Erazo  
xavierguambo@hotmail.com

---

David Rafael Morante Lombeida  
david\_korn90@hotmail.com

## RESUMEN

El Sistema MyVPN, fue desarrollado enfocándonos en las necesidades de las medianas empresas, específicamente en la implementación de Redes Privadas Virtuales a un bajo costo, todo esto posible gracias a que solo utilizamos herramientas Open Source y de libre distribución en la Web.

Desarrollado en el lenguaje PHP, con un ambiente amigable e intuitivo, la navegación por el sistema se convierte en tarea sencilla, ya que cuenta con Configuración Asistida y Automática, que guiará al usuario a efectuar una correcta configuración de la VPN en los protocolos PPTP y L2TP /IPSEC.

Le recordamos que nuestro sistema dará una IP a los clientes VPN, de la Red Principal de la LAN configurada en su servidor Linux. Los servicios que pudieran implementarse en esta interface LAN, tales como SAMBA, FTP, etc., dependerá del administrador del servidor.

## RESUME

MyVPN System is built by a process of what medium and small business need, focusing in the VPN (Virtual PrivateNetwork) Implementations, we decided to build this system to a low cost, and with all the basic features that have been needed on the VPN. Their low costs it's possible because we use only open source tools of free distribution on the web.

MyVPN System built in PHP, it provides a friendly user interface that will help to users to understand easily how it works. Furthermore good VPN configurations will not any more a big problem by choosing "The Automatic Wizard VPN Configuration". Also the user has the choice to do this configuration with the "Custom Wizard"

We ought to tell you, MyVPN System, like most VPN software, will give to VPN clients an IP of the Linux Sever LAN interface. All services that could be work in that interface like Samba, FTP, etc., depend that how the administrator would to implement it.

## ÍNDICE GENERAL

|   |          |
|---|----------|
| AGRADECIMIENTOS.....                          | II       |
| DEDICATORIAS.....                             | V        |
| TRIBUNAL DE GRADUACIÓN.....                   | VIII     |
| DECLARACION EXPRESA.....                      | IX       |
| RESUMEN.....                                  | X        |
| INDICE GENERAL.....                           | XI       |
| <br>  |          |
| <b>CAPITULO 1 .....</b>                       | <b>1</b> |
| <b>INTRODUCCIÓN.....</b>                      | <b>1</b> |
| <b>1.1. PROBLEMÁTICA .....</b>                | <b>2</b> |
| <b>1.2. SOLUCIÓN.....</b>                     | <b>3</b> |
| <b>1.3. JUSTIFICACIÓN .....</b>               | <b>4</b> |
| <b>1.3.1. ¿POR QUÉ LINUX? .....</b>           | <b>4</b> |
| <b>1.3.2. ¿POR QUÉ VPN?.....</b>              | <b>5</b> |
| <b>1.3.3. TIPO DE CONEXIÓN DE LA VPN.....</b> | <b>5</b> |
| <b>1.4. MISIÓN.....</b>                       | <b>6</b> |
| <b>1.5. VISIÓN .....</b>                      | <b>6</b> |
| <b>1.6. OBJETIVOS GENERALES.....</b>          | <b>7</b> |

|  |           |
|--|-----------|
| <b>1.7. OBJETIVOS ESPECÍFICOS .....</b>                        | <b>8</b>  |
| <b>1.8. ALCANCES .....</b>                                     | <b>9</b>  |
| <b>1.9. RECURSOS DE HARDWARE Y SOFTWARE .....</b>              | <b>10</b> |
| <b>1.9.1. HARDWARE .....</b>                                   | <b>10</b> |
| <b>1.9.2. DISTRIBUCIÓN DE EQUIPOS.....</b>                     | <b>11</b> |
| <b>1.9.3. SOFTWARE. ....</b>                                   | <b>11</b> |
| <b>1.9.3.1. SISTEMAS OPERATIVOS.....</b>                       | <b>11</b> |
| <b>1.9.3.2. HERRAMIENTA DE DESARROLLO.....</b>                 | <b>12</b> |
| <b>1.9.3.3. BASE DE DATOS.....</b>                             | <b>12</b> |
| <b>1.9.3.4. UTILITARIOS.....</b>                               | <b>12</b> |
| <b>1.10. SERVICIOS QUE INSTALAREMOS EN EL SERVER VPN .....</b> | <b>12</b> |
| <b>1.10.1. APACHE .....</b>                                    | <b>12</b> |
| <b>1.10.2. BIND (BERKELEY INTERNET NAME DOMAIN).....</b>       | <b>13</b> |
| <b>1.10.3. POSTGRESQL.....</b>                                 | <b>15</b> |
| <b>1.11. RECURSO HUMANO.....</b>                               | <b>15</b> |
| <b>1.11.1. ANALISTA DE INFRAESTRUCTURA. ....</b>               | <b>16</b> |
| <b>1.11.2. ANALISTA DE SISTEMAS.....</b>                       | <b>16</b> |
| <b>1.11.3. ANALISTA PMO .....</b>                              | <b>16</b> |
| <b>1.12. MODELO DE DESARROLLO.....</b>                         | <b>17</b> |
| <b>1.12.1. MODELO DEL PROTOTIPO ESPIRAL. ....</b>              | <b>17</b> |
| <b>1.12.2. EXPLICANDO CADA FASE TENEMOS.....</b>               | <b>18</b> |
| <b>1.13. METODOLOGÍA. ....</b>                                 | <b>18</b> |

|   |           |
|---|-----------|
| 1.14. ESTUDIO DE FACTIBILIDAD.....                        | 18        |
| 1.14.1. FACTIBILIDAD TÉCNICA.....                         | 19        |
| 1.14.2. FACTIBILIDAD ECONÓMICA .....                      | 19        |
| 1.14.3. FACTIBILIDAD OPERACIONAL .....                    | 19        |
| 1.15. ANÁLISIS FODA.....                                  | 20        |
| <b>CAPITULO 2 .....</b>                                   | <b>23</b> |
| <b>ANÁLISIS .....</b>                                     | <b>23</b> |
| 2.1 LEVANTAMIENTO DE LA INFORMACIÓN. ....                 | 24        |
| 2.2 ANÁLISIS DE LOS REQUERIMIENTOS Y PRERREQUISITOS. .... | 25        |
| 2.2.1 PRERREQUISITOS.....                                 | 25        |
| 2.2.2 REQUERIMIENTOS .....                                | 25        |
| 2.3 ESTUDIO DE FACTIBILIDAD.....                          | 26        |
| 2.3.1 FACTIBILIDAD TÉCNICA .....                          | 27        |
| 2.3.2 FACTIBILIDAD ECONÓMICA .....                        | 28        |
| 2.3.3 FACTIBILIDAD OPERACIONAL .....                      | 28        |
| 2.4 DIAGRAMA DE FLUJO DE DATOS .....                      | 30        |
| 2.5 DIAGRAMA DE CASOS DE USO .....                        | 37        |
| <b>CAPITULO 3 .....</b>                                   | <b>41</b> |
| <b>DISEÑO .....</b>                                       | <b>41</b> |
| 3.1 SERVIDOR .....  | 41        |
| 3.1.1 INGRESO .....                                       | 42        |

|                                       |  |    |
|---------------------------------------|--|----|
| 3.1.2                                 | VENTANA INGRESO .....  | 42 |
| 3.1.3                                 | CONFIGURACIÓN DE VPN.....                                      | 43 |
| 3.1.4                                 | TIPO DE CONFIGURACIÓN.....                                     | 44 |
| 3.1.5                                 | CONFIGURACIÓN AUTOMÁTICA.....                                  | 45 |
| 3.1.6                                 | ASIGNACIÓN DE IP .....   | 46 |
| 3.1.7                                 | CREACIÓN DE USUARIOS.....                                      | 47 |
| 3.1.7.1                               | CREACIÓN DE USUARIO DENTRO DE LA CONFIGURACIÓN AUTOMÁTICA. ... | 48 |
| 3.1.7.2                               | CREACIÓN DE USUARIO EN MANTENIMIENTO DE USUARIO.....           | 48 |
| 3.1.8                                 | MANTENIMIENTO.....   | 48 |
| 3.1.9                                 | OPCIONES DEL SERVIDOR .....                                    | 50 |
| 3.1.10                                | CUENTAS DE USUARIO .....                                       | 50 |
| 3.1.11                                | CONEXIONES ACTIVAS .....                                       | 51 |
| 3.1.12                                | CONEXIONES DEL DÍA .....                                       | 52 |
| 3.2                                   | CLIENTE.....   | 54 |
| 3.2.1                                 | INGRESO .....  | 54 |
| 3.2.2                                 | VENTANA DE CONEXIÓN PARA CLIENTES VPN.....                     | 55 |
| 3.2.3                                 | AJUSTES ADICIONALES PARA CLIENTES L2TP/IPSEC.....              | 55 |
| 3.2.3.1                               | SEGURIDAD .....  | 56 |
| 3.2.3.2                               | FUNCIONES DE RED .....   | 57 |
| 3.3                                   | DISEÑO ARQUITECTÓNICO .....                                    | 59 |
| CAPITULO 4 .....                      |  | 61 |
| DESARROLLO Y PRUEBA DEL SISTEMA ..... |  | 61 |

|  |           |
|--|-----------|
| <b>4.1. DESARROLLO DEL SISTEMA .....</b>               | <b>61</b> |
| <b>4.1.1. CREACIÓN DE BASE DE DATOS .....</b>          | <b>62</b> |
| <b>4.1.1.1. DESCRIPCIÓN DE LOS CAMPOS .....</b>        | <b>63</b> |
| <b>4.1.1.1.1. USERS .....</b>                          | <b>63</b> |
| <b>4.1.1.1.2. CONEXIONES.....</b>                      | <b>64</b> |
| <b>4.1.2. CREACIÓN DE LOS COMPONENTES .....</b>        | <b>65</b> |
| <b>4.1.2.1. AUTENTICACIÓN .....</b>                    | <b>66</b> |
| <b>4.1.2.2. ESCANEEO .....</b>                         | <b>66</b> |
| <b>4.1.2.3. ANÁLISIS DE DATOS .....</b>                | <b>66</b> |
| <b>4.1.2.4. ALMACENAMIENTO DE DATOS.....</b>           | <b>66</b> |
| <b>4.1.2.5. CAMBIOS SUGERIDOS POR EL USUARIO .....</b> | <b>67</b> |
| <b>4.1.2.6. REPORTE.....</b>                           | <b>67</b> |
| <b>4.1.3. SEGURIDADES.....</b>                         | <b>67</b> |
| <b>4.1.3.1. SISTEMA OPERATIVO.....</b>                 | <b>68</b> |
| <b>4.1.3.2. BASE DE DATOS.....</b>                     | <b>69</b> |
| <b>4.1.3.3. HUMANO.....</b>                            | <b>69</b> |
| <b>4.2. PRUEBAS DEL SISTEMA.....</b>                   | <b>69</b> |
| <b>4.2.1. VERIFICACIÓN Y VALIDACIÓN.....</b>           | <b>70</b> |
| <b>4.2.2. PRUEBAS DE UNIDAD .....</b>                  | <b>70</b> |
| <b>4.2.2.1. AUTENTICACIÓN .....</b>                    | <b>71</b> |
| <b>4.2.2.2. ANÁLISIS .....</b>                         | <b>71</b> |
| <b>4.2.2.3. REPORTE.....</b>                           | <b>71</b> |

|   |           |
|---|-----------|
| 4.2.2.4. PRUEBAS DE INTEGRACIÓN .....                   | 72        |
| <b>CAPITULO 5 .....</b>                                 | <b>73</b> |
| <b>IMPLEMENTACIÓN DEL SISTEMA .....</b>                 | <b>73</b> |
| 5.1 ELEMENTOS FÍSICOS.....                              | 75        |
| 5.2 ELEMENTOS LÓGICOS .....                             | 76        |
| 5.3 ELEMENTOS HUMANOS .....                             | 77        |
| 5.4 INFRAESTRUCTURA .....                               | 77        |
| <b>CAPITULO 6 .....</b>                                 | <b>78</b> |
| <b>RECOMENDACIONES Y CONCLUSIONES DE LA TESIS .....</b> | <b>78</b> |
| 6.1. RECOMENDACIONES .....                              | 78        |
| 6.1.1. HARDWARE .....                                   | 79        |
| 6.1.2. SOFTWARE .....                                   | 79        |
| 6.2. CONCLUSIONES .....                                 | 80        |
| <b>BIBLIOGRAFÍA .....</b>                               | <b>81</b> |

## ÍNDICE DE GRÁFICOS

|  |           |
|--|-----------|
| <b>GRÁFICO 1 - SERVIDOR VPN CON CLIENTES LINUX Y WINDOWS .....</b> | <b>3</b>  |
| <b>GRÁFICO 2 - TÚNEL DE CONEXIÓN VPN .....</b>                     | <b>6</b>  |
| <b>GRÁFICO 3 - MODELO DE PROTOTIPO ESPIRAL .....</b>               | <b>17</b> |
| <b>GRÁFICO 4A - FLUJO DE DATOS DE MYVPN NIVEL 0.....</b>           | <b>32</b> |
| <b>GRÁFICO 4B - FLUJO DE DATOS DE MYVPN .....</b>                  | <b>33</b> |
| <b>GRAFICO 4C - CONFIGURACIÓN DE PPTP .....</b>                    | <b>34</b> |
| <b>GRÁFICO 4D - CONFIGURACIÓN DE L2TP/IPSEC.....</b>               | <b>35</b> |
| <b>GRAFICO 4E - ESTRUCTURA DE USUARIOS EN LÍNEA .....</b>          | <b>36</b> |
| <b>GRÁFICO 5A - CASO DE USO (ADMINISTRADOR).....</b>               | <b>38</b> |
| <b>GRÁFICO 5B - CASO DE USO (CLIENTES PPTP).....</b>               | <b>39</b> |
| <b>GRÁFICO 5C - CASO DE USO (CLIENTES L2TP/IPSEC).....</b>         | <b>40</b> |
| <b>GRÁFICO 6 - VENTANA DE INGRESO.....</b>                         | <b>42</b> |
| <b>GRÁFICO 7 - VENTANA DE CONFIGURACIÓN .....</b>                  | <b>43</b> |
| <b>GRÁFICO 8 - TIPO DE CONFIGURACIÓN .....</b>                     | <b>44</b> |
| <b>GRÁFICO 9 - VENTANA DE PREVIA CONFIGURACIÓN AUTOMÁTICA.....</b> | <b>45</b> |
| <b>GRÁFICO 10 - VENTANA DE IP ASIGNADAS POR DEFAULT .....</b>      | <b>46</b> |

|   |           |
|---|-----------|
| <b>GRÁFICO 11 - VENTANA DE CREACIÓN DE USUARIOS .....</b>                                   | <b>47</b> |
| <b>GRÁFICO 12 - VENTANA DE MANTENIMIENTO .....</b>  | <b>49</b> |
| <b>GRÁFICO 13 - OPCIONES DE SERVIDOR .....</b>  | <b>50</b> |
| <b>GRÁFICO 14 - CUENTAS DE USUARIO .....</b>  | <b>51</b> |
| <b>GRÁFICO 15 - CUENTAS ACTIVAS .....</b>   | <b>52</b> |
| <b>GRÁFICO 16 - CONEXIONES DEL DÍA .....</b>  | <b>53</b> |
| <b>GRÁFICO 17 - VENTANA DE INGRESO DE USUARIO Y CONTRASEÑA EN CLIENTES<br/>WINDOWS.....</b> | <b>55</b> |
| <b>GRÁFICO 18 - PESTAÑA DE SEGURIDAD EN PROPIEDADES DE CONEXIÓN .....</b>                   | <b>56</b> |
| <b>GRÁFICO 19 - INGRESO DE CLAVE COMPARTIDA .....</b>                                       | <b>57</b> |
| <b>GRÁFICO 20 - SELECCIÓN DE VPN CON L2TP/IPSEC EN PROPIEDADES DE<br/>CONEXIÓN .....</b>    | <b>58</b> |
| <b>GRAFICO 21A - DISEÑO ARQUITECTÓNICO (VALIDACIÓN USUARIO) .....</b>                       | <b>59</b> |
| <b>GRAFICO 21B - DISEÑO ARQUITECTÓNICO (ASIGNACIÓN IP).....</b>                             | <b>59</b> |
| <b>GRAFICO 21C - DISEÑO ARQUITECTÓNICO (GESTIÓN DE CONFIGURACIÓN) .....</b>                 | <b>60</b> |

## ÍNDICE DE TABLAS

|   |           |
|---|-----------|
| <b>TABLA 1 - HARDWARE REQUERIDO .....</b>                                 | <b>11</b> |
| <b>TABLA 2 - RECURSO HUMANO .....</b>                                     | <b>16</b> |
| <b>TABLA 3 - PRECIOS DE VARIOS DISPOSITIVOS DE RED Y COMPUTADOR .....</b> | <b>27</b> |
| <b>TABLA 4 - COSTOS PARA LA REALIZACIÓN DEL APLICATIVO .....</b>          | <b>28</b> |
| <b>TABLA 5A - TABLA "USERS" .....</b>                                     | <b>62</b> |
| <b>TABLA 5B - TABLA "CONEXIONES" .....</b>                                | <b>63</b> |

# **CAPITULO 1**

## **INTRODUCCIÓN**

Las distancias se han acortado en estos días gracias a Internet. La gran parte de nosotros enviamos cualquier archivo por la Web a través del mail, ftp, etc.; pero, ¿qué pasa si deseo enviar información sumamente privada a través de la Web? Por ejemplo, las nóminas de pago de alguna institución, claves de tarjetas de crédito, el monto de las ventas del mes. De seguro que la persona dueña de esta información estaría muy preocupada si su información fuera tomada por terceros que podrían perjudicarlo. Una solución segura a este problema sería la implementación de una Red Virtual Privada (VPN).

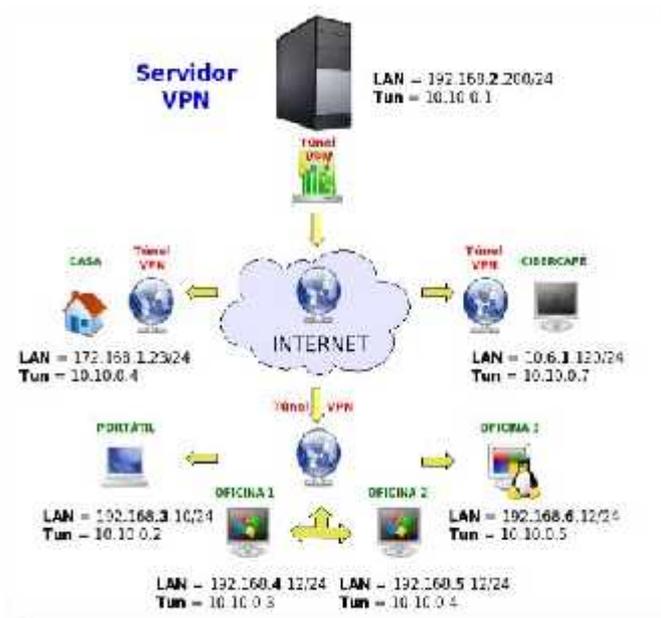
Ahora, tomemos en consideración de que por lo general esa persona que desea la transmisión de información, trabaja en una empresa o es dueño de la misma y, en dicha empresa tiene sus propias configuraciones tales como: Determinada impresora para documentos blanco y negro y otra para los documentos de color, las carpetas compartidas, etc. También tomemos en cuenta que cada empresa con su respectiva red tiene una configuración específica para sus usuarios (perfil de usuario), con atributos y características por cada tipo de grupo (según su necesidad).

### **1.1. Problemática**

Hoy en día, muchas empresas no se ubican solamente en un lugar físico, hay varias sucursales (ramificaciones) de la misma, como por ejemplo, sucursales de puntos de ventas o de recepción de mercadería, sucursales que ofrecen otro tipo de servicio pero que pertenece a la misma empresa, en fin, es infinita las circunstancias por las que una empresa necesita colocar una extensión de sus servicios, en un lugar diferente de su matriz y si ésta funciona con una tecnología implementada a nivel informático (sistemas operativos, software de inventario, recursos humanos, bases de datos, etc), no es difícil prever que la extensión o ramificación de esta empresa necesite trabajar o interactuar con los recursos de su *red principal* y tener todos los beneficios que le proporciona normalmente en la matriz. ¿Cómo lograr aquello?

## 1.2. Solución

Una de las maneras más fiables de comunicación remota son las redes privadas virtuales (VPN). Se les llama privadas porque se establecen exclusivamente entre el receptor y emisor de la información. Son virtuales porque no se necesita un cable o cualquier otro medio físico directo entre los comunicantes, la sesión se establece usando una infraestructura pública, por lo tanto, es una red no física privada que viaja sobre una pública y virtualmente extiende la red de la empresa hasta donde se encuentre el cliente. Como acotación aclaramos que una Red Privada no es lo mismo que una Red Privada Virtual. La primera utiliza líneas alquiladas para formar toda la Red Privada. La VPN lo que hace es crear un túnel entre los dos puntos a conectar utilizando infraestructura pública.



**Gráfico 1** - Servidor VPN con clientes Linux y Windows

Las redes privadas virtuales crean un túnel o conducto de un sitio a otro para transferir datos a esto se le conoce como encapsulación, además los paquetes van encriptados de forma que los datos son ilegibles para los extraños.

### **1.3. Justificación**

La VPN, como una tecnología informática, se puede implementar en varias plataformas del mercado existente a parte de la conocida Windows. Sin embargo, para la realización de este proyecto hemos decidido implementarlo bajo Linux.

#### **1.3.1. ¿Por qué Linux?**

Se dice que las mejores cosas de la vida no cuestan, y es cierto. Pero cuando se trata de un negocio, el empresario no puede arriesgarse a ciegas, y por eso se pregunta y asesora para conocer qué tan bueno es un sistema operativo de libre distribución. Esta es la cuestión principal: ¿por qué usar Linux en lugar de otros sistemas operativos que tienen más tiempo en operación? La respuesta es sencilla, porque Linux es un sistema operativo con muchas ventajas: confiabilidad probada, la mejor seguridad, la más poderosa funcionalidad,

administración sencilla, el mejor desempeño y, en especial el más bajo costo.

### 1.3.2. ¿Por qué VPN?

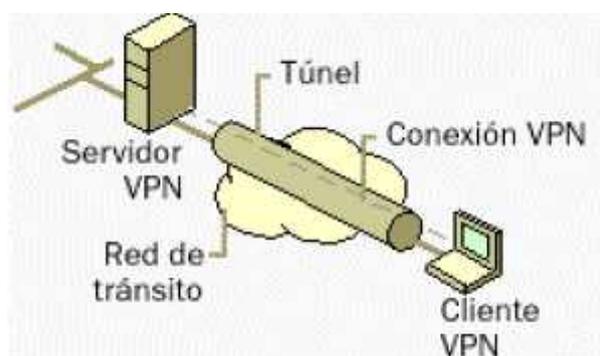
- ✓ **Seguridad**: provee encriptación y encapsulación de datos de manera que hace que estos viajen codificados y a través de un túnel.
- ✓ **Costos**: ahorran grandes sumas de dinero en líneas dedicadas o enlaces físicos.
- ✓ **Mejor administración**: cada usuario que se conecta puede tener un número de IP fija asignada por el administrador, lo que facilita algunas tareas como por ejemplo: conectarse al servidor de correo de la compañía, ejecutar algún aplicativo propio de la intranet de la compañía, etc.
- ✓ **Facilidad para los usuarios** con poca experiencia para conectarse a grandes redes corporativas transfiriendo sus datos de forma segura.

### 1.3.3. Tipo de conexión de la VPN

#### **Conexión de acceso remoto (host a cliente)**

Una conexión de acceso remoto es realizada por un cliente o un usuario de un computador que se conecta a una red

privada, los paquetes enviados a través de la conexión VPN son originados al cliente de acceso remoto, y este se autentica al servidor de acceso remoto, y el servidor se autentica ante el cliente.



**Gráfico 2** - Túnel de Conexión VPN

#### **1.4. Misión**

Elaborar un sistema en un ambiente web con herramientas netamente open source (código abierto), la cual pueda ayudar a configurar de manera sencilla las herramientas necesarias para la implementación de una Red Privada Virtual y sus clientes.

#### **1.5. Visión**

Fortalecer el uso de VPN en las medianas empresas, las mismas que acorten distancias, abaraten costos, sean seguras y que, nuestro

aplicativo sea un ejemplo más de la eficacia y eficiencia en el desarrollo de soluciones open source.

### **1.6. Objetivos Generales**

Hoy en día existen varios Hardware que podrían realizar una conexión VPN de manera sencilla, con todas sus respectivas configuraciones de Host, tipos de Usuario y Reglas. Uno de los más difundidos es el de la Tecnología CISCO Pix. Es conocido debido a su fácil instalación y configuración y una amplia gama de parametrización de reglas (como cierre de puertos, firewall, etc).

Las VPN representan una gran solución para las empresas en cuanto a cumplir con los principios de la Seguridad de la Información (disponibilidad, integridad y confidencialidad) y prácticamente se ha vuelto un tema importante en las organizaciones, debido a que reduce significativamente el costo de la transferencia de datos de un lugar a otro, el único inconveniente que pueden tener las VPN son: primero, se deben establecer correctamente las políticas de seguridad y segundo, de acceso, porque si esto no está bien definidas las políticas de de acceso (Firewall) pueden suscitarse

consecuencias serias como ataques al Servidor o robo de información.

- ✓ Nuestro objetivo será el desarrollar un aplicativo (en un servidor Linux) que configure los servicios necesarios para el levantamiento de una VPN y también la creación de clientes remotos VPN. Así una persona fuera de nuestra red principal, podrá conectarse a ella desde un sitio remoto, y operar en la red como si estuviera ahí.

### **1.7. Objetivos Específicos**

- ✓ Una de las principales bondades de la instalación VPN es que puede interactuar con sistemas Windows, Linux o Mac. Nuestro objetivo principal será trabajar con usuarios Windows y Linux.
- ✓ Una aplicación Web de fácil parametrización para la creación y eliminación de usuarios tipo VPN en el Servidor Linux.
- ✓ El aplicativo facilitará los valores de las interfaces de Red (IP, Máscara, Gateway) con el fin de ayudar al administrador del sistema a configurar correctamente la VPN.

## 1.8. Alcances

Nuestro aplicativo se implementará en un ambiente web desarrollado bajo el lenguaje PHP v.5 y; como base de datos, se implementará en Postgres. Estas dos herramientas son Open Source y compatibles con Linux.

Nuestro aplicativo contará con las siguientes bondades:

1. Verificación de una correcta configuración de la Red Principal, es decir, mostrará los valores de IP, Máscara y Puerta de Enlace tanto de la interface WAN como de la LAN.
2. Un “wizard de configuración automática de VPN”, para que un usuario novato pueda levantar los servicios necesarios para su implementación de manera eficaz y sencilla.
3. Establecer una conexión VPN con los protocolos PPTP (Point to Point Tunneling Protocol) que es desarrollado por Microsoft; con el protocolo L2TP (Layer 2 Tunneling Protocol) sobre IPsec (OpenSwan).
4. Autenticación de usuarios a través del protocolo CHAP (*Challenge Handshake Authentication Protocol*), siendo este protocolo un método de autenticación usado por servidores accesibles vía PPTP. CHAP verifica periódicamente la identidad del cliente remoto usando un intercambio de información de tres etapas. Esto ocurre cuando se establece el

enlace inicial y puede pasar de nuevo en cualquier momento de la comunicación. La verificación se basa en un secreto compartido (como una contraseña). Esta autenticación se usará para la implementación de VPN con los protocolos PPTP y L2TP/IPSec.

5. Monitoreo de la navegación de los usuarios VPN en línea, teniendo información básica del usuario e IP con que se conecta.
6. Bitácora de usuarios que se conectaron durante el transcurso del día, así estos solo se hayan conectado por minutos.
7. Realizar pruebas de impresión, así también como compartir archivos con integridad, confidencialidad y seguridad.

## **1.9. Recursos de Hardware y Software**

### **1.9.1. Hardware**

Para el desarrollo del portal Web se va a necesitar los siguientes equipos:

| PC'S | CARACTERÍSTICAS BÁSICAS                      |
|------|--|
| 3    | Procesador Intel P4 o AMD Athlom XO de 2 Ghz |
| 3    | 1 Gb Ram recomendado                         |
| 3    | Disco Duro de 80 Gb                          |
| 3    | Monitor de 15                                |
| 3    | Unidad DVD-RW                                |
| 1    | Router                                       |
| 1    | Switch                                       |

**Tabla 1 - Hardware Requerido**

### 1.9.2. Distribución de equipos.

1 PC que se utilizara como servidor de VPN

1 PC que se utilizara para el desarrollo de la herramienta

1 PC que se usara como equipo cliente al cual se le dará acceso por medio de la VPN.

### 1.9.3. Software.

#### 1.9.3.1. Sistemas Operativos.

En el Servidor VPN se instalará el sistema operativo Linux Centos v.5.0 con kernel 2.6.2, por cuanto tiene todos los servicios a nivel de servidores que requerimos para nuestro proyecto.

En el equipo cliente se instalará el sistema operativo Windows XP con SP2, ya que es el más difundido actualmente en el mercado.

#### **1.9.3.2. Herramienta de Desarrollo.**

PHP v. 5.0

Apache HTTP V. 2.0

Macromedia Dreamweaver 8

Komodo Edit 4.4 (Editor PHP para Linux)

PgAdmin3-1.4

#### **1.9.3.3. Base de Datos.**

PostgreSQL 8.1.11

#### **1.9.3.4. Utilitarios.**

Microsoft Office, WinZip, Openoffice.

### **1.10. Servicios que instalaremos en el Server VPN**

#### **1.10.1. Apache**

Apache es un servidor web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos (HTTP 1.1). Entre sus características destacan:

- Multiplataforma.
- Es un servidor de web conforme al protocolo HTTP/1.1
- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- Se desarrolla de forma abierta.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, el lenguaje de programación que utilizaremos para el desarrollo de nuestro aplicativo en ambiente web.

#### **1.10.2. Bind (Berkeley Internet Name Domain).**

BIND (acrónimo de Berkeley Internet Name Domain) es una implementación del protocolo DNS y provee una implementación libre de los principales componentes del Sistema de Nombres de Dominio, los cuales incluyen:

- Un servidor de sistema de nombres de dominio (named).
- Una biblioteca resolutoria de sistema de nombres de dominio
- Herramientas para verificar la operación adecuada del servidor DNS (bind-utils).

El Servidor DNS BIND es ampliamente utilizado en la Internet (99% de los servidores DNS) proporcionando una robusta y estable solución.

DNS (acrónimo de Domain Name System) es una base de datos distribuida y jerárquica que almacena la información necesaria para los nombre de dominio. Sus usos principales son la asignación de nombres de dominio a direcciones IP y la localización de los servidores de correo electrónico correspondientes para cada dominio.

Los Servidores DNS utilizan TCP y UDP en el puerto 53 para responder las consultas. Casi todas las consultas consisten de una sola solicitud UDP desde un Cliente DNS seguida por una sola respuesta UDP del servidor.

Una correcta configuración del DNS será obligación del Administrador del Servidor Linux, ya que es necesario para el buen funcionamiento de la configuración de la VPN.

### **1.10.3. Postgresql**

PostgreSQL es una base de datos relacional, distribuida bajo licencia BSD y con su código fuente disponible libremente. Es el motor de bases de datos de código abierto más potente del momento y en sus últimas versiones empieza a no tener que envidiarle nada a otras bases de datos comerciales.

Utilizaremos la base de datos Postgresql con fines muy sencillos y prácticos; primero, para ingreso y cambio de clave del administrador del sistema y, segundo, para el ingreso de auditoría de usuarios conectados en el día, si es que el administrador lo requiere.

### **1.11. Recurso Humano**

Contaremos con la participación de los 3 integrantes del grupo, los cuales se desempeñan de la siguiente manera:

| <b>Recurso Humano</b> | <b>Rol o Perfil</b>         | <b>Nombres</b>      |
|-----------------------|-----------------------------|---------------------|
| 1                     | Analista de Infraestructura | Miguel Contreras J. |
| 1                     | Analista de Sistemas        | Xavier Guambo E.    |
| 1                     | Analista PMO                | David Morante       |

**Tabla 2 - Recurso Humano**

**1.11.1. Analista de Infraestructura.**

Investigación, implementación de nuevas tecnologías (hardware y software).

**1.11.2. Analista de Sistemas.**

Sera responsable de realizar el análisis y esquema a seguir sobre el modelado y estándares a utilizar.

**1.11.3. Analista PMO**

PMO significa oficina de Gestion de Proyectos, sera responsable de garantizar el cumplimiento de la metodologia de gestion de proyectos.

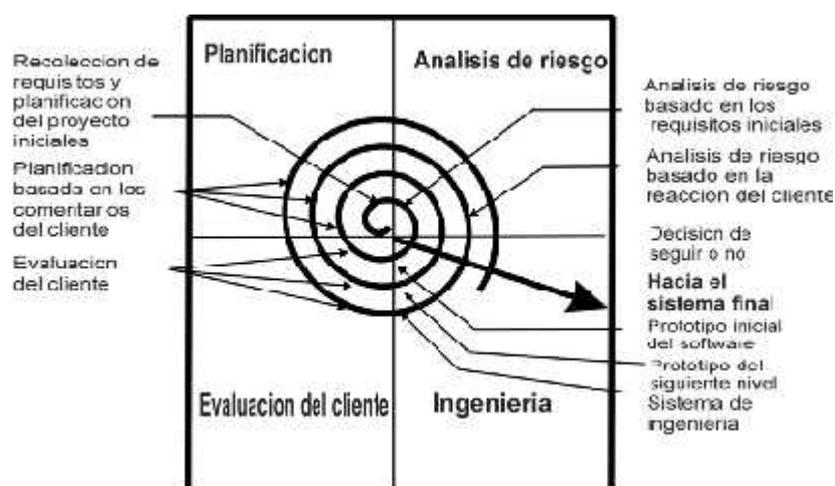
## 1.12. Modelo de Desarrollo.

Las actividades de este modelo son una espiral, cada bucle es una actividad en el desarrollo de nuestro aplicativo. Las actividades no están fijadas a prioridad, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior.

En el análisis de riesgo abarcaremos los riesgos potenciales y seleccionaremos una o varias alternativas propuestas para reducir o eliminar dichos riesgos.

### 1.12.1. Modelo del prototipo Espiral.

El sistema en su totalidad está basado en el modelo de prototipo Espiral que detallamos a continuación.



**Gráfico 3 - Modelo de prototipo Espiral**

### **1.12.2. Explicando cada fase tenemos.**

- ✓ Planificación: determinación de objetivos, alternativas y restricciones.
- ✓ Análisis de riesgo: análisis de alternativas e identificación/resolución de riesgos.
- ✓ Ingeniería: desarrollo del producto del "siguiente nivel",
- ✓ Evaluación del cliente: Valorización de los resultados de la ingeniería.

### **1.13. Metodología.**

La metodología usada para el desarrollo de nuestro aplicativo es la Estructurada o Clásica ya que el mismo no requiere necesariamente de diseño orientado a objetos, porque no es un software demasiado complejo pero si requeriremos aplicar una metodología que incluya los diseños básicos y que cumplan con el Ciclo de Vida de un Sistema.

### **1.14. Estudio de Factibilidad**

Para el estudio de factibilidad de nuestro proyecto, tomamos en cuenta 3 clásicos aspectos: el técnico, económico y operativo.

#### **1.14.1. Factibilidad Técnica**

El proyecto es totalmente realizable desde el punto de vista técnico ya que existen en el mercado actual las herramientas tecnológicas para implementarlo, tales como router, switch, cableado estructurado, software requerido. Algunos de estos dispositivos los mencionaremos con más detalle en el capítulo 2 de Análisis.

#### **1.14.2. Factibilidad económica**

El software de distribución de Linux (Centos 5) es gratis, así como también el lenguaje PHP, la base datos Postgress y el Servidor de Páginas Web Apache; lo cual es una gran ventaja. Sin embargo, el recurso humano, el tiempo y la infraestructura requeridos para nuestro aplicativo tienen un costo estimado, el cual lo detallaremos en el siguiente capítulo.

#### **1.14.3. Factibilidad Operacional**

Nuestro aplicativo va dirigido a personas del área de Sistemas con conocimientos básicos de redes, así que el impacto que podría suscitarse, al implementar esta nueva tecnología en la empresa, será mínimo, debido a que el aplicativo contará con

un ambiente web agradable, descriptivos botones e intuitivas páginas para que la administración de la red VPN sea mucho más fácil.

Como nuestro aplicativo podría manejarlo una persona con conocimientos básicos de redes, se puede prever que el mismo va a dirigido para empresas pequeñas y medianas, ya que por lo general estas empresas no necesariamente contratan a un experto en redes informáticas para el mantenimiento de sus redes, debido al presupuesto limitado con que se manejan. En el siguiente capítulo detallaremos un pequeño muestreo de cómo las empresas manejan el área informática de acuerdo al número de computadores con los que cuentan.

#### **1.15. Análisis FODA**

##### **FORTALEZAS**

- ✓ Bajo costo ya que se implementa en un software libre y con licencias GPL.
- ✓ Fácil de usar, gracias al ayudante “Wizard de Configuración de Configuración Automática de VPN” que implementa el aplicativo.

- ✓ Escalabilidad en la red principal, es decir, los servicios de nuestra red podrán extenderse fuera de nuestra oficina a un bajo costo.
- ✓ Seguridad en la transmisión de datos si se utiliza el protocolo L2TP/IPSec.

### **OPORTUNIDADES**

- ✓ Poca implementación de VPN en medianas empresas, por lo general utilizan FTP.
- ✓ Adaptabilidad y flexibilidad de la VPN ya que permite una conexión remota entre la empresa y sucursales, entre la empresa y sus socios, proveedores, etc.
- ✓ Difusión rápida del aplicativo por su licencia libre.

### **DEBILIDADES**

- ✓ Se necesita Internet tanto del lado del Servidor como del cliente para su conexión.
- ✓ El protocolo PPTP ha sido vulnerado en algunas ocasiones por ciertos hackers.
- ✓ Existencia de muchos aplicativos en el mercado que implementan VPN.

- ✓ Se necesita de una persona con conocimientos medianos en redes LAN y WAN.

### **AMENAZAS**

- ✓ El intento de robo de información por terceros (hackers), ya que a pesar de que los datos viajan encriptados por un túnel a través de internet, los hackers siempre están buscando nuevas formas de acceder a la información.

# **CAPITULO 2**

## **ANÁLISIS**

En el presente capítulo desarrollaremos la fase de “Análisis del Sistema” a la implementación de nuestra VPN sobre Linux. De manera cronológica nos introduciremos en las funciones del software, el comportamiento del mismo hacia agentes externos, modelos jerárquicos y hardware. Se lo realiza de esta forma para que la información presentada sea de fácil entendimiento para cualquier persona que desee guiarse del presente documento.

## **2.1 Levantamiento de la información.**

Esta etapa se la realizó mediante dos vías: las encuestas y revisión de documentos por internet. En las encuestas, se recopiló la información de las principales necesidades de una empresa al querer implementar una VPN; el factor de seguridad y economía fueron los más importantes. La información que se realizó por internet arrojó como resultado que existen un sin número de herramientas que implementan tecnología VPN y que en Ecuador, en su mayoría sólo las grandes empresas tienen implementado ésta tecnología. También se determinó que las soluciones de CISCO Plix, son las más difundidas para el levantamiento de VPN.

Para ofrecer un excelente aplicativo, se evaluó las características de las diferentes implementaciones de VPN que ofrecen en el mercado, con el fin de acoger a nuestro proyecto las tareas más comunes y requeridas por el usuario, brindándole un producto diferente a un costo bajo pero con similares funcionalidades.

## **2.2 Análisis de los requerimientos y prerequisites.**

### **2.2.1 Prerequisites**

#### **SERVIDOR**

Para la instalación del aplicativo, nos aseguraremos que el Servidor tenga bien definidos sus parámetros de red y en especial el DNS; esto es esencial ya que una red principal de LAN debe contar con identidad propia, es decir, un dominio; como por ejemplo: midominio.com.

Para que se pueda establecer el túnel VPN, deberá contar con servicio de Internet a través de una IP Pública.

#### **CLIENTE**

Para el cliente de plataforma Windows, sólo se necesitará que tenga habilitada la tarjeta de red; una IP Pública, para la comunicación remota con el Servidor VPN

### **2.2.2 Requerimientos**

Para el desarrollo de nuestro aplicativo necesitaremos investigar:

Cuáles son los archivos (en mi servidor Linux) para configurar y fijar la IP, Máscara, Gateway y DNS.

Configuración de Iptables para permisos de acceso de clientes remotos al Servidor VPN.

Puertos con los que trabaja los protocolos PPTP, L2TP/ IPsec.

Cómo funciona la autenticación CHAP v.2.

Cómo encriptar los datos que viajarán por nuestro túnel de conexión VPN y la forma de encriptación por cada protocolo de conexión (PPTP, L2TP/IPsec)

Cómo asignar un ancho de banda fijo para los clientes remotos VPN.

Los registros (archivos log) donde se guardan los sucesos de conexión de los usuarios remotos VPN.

La forma de manejar la contraseña del Administrador de forma segura o encriptada.

Cómo instalar en el Servidor Linux el Apache para el alojamiento de las páginas web de nuestro aplicativo.

El lenguaje PHP para la elaboración de páginas web.

### **2.3 Estudio de Factibilidad**

En el capítulo anterior mencionamos que nuestro aplicativo es completamente factible en los 3 aspectos más importantes para la implementación de un proyecto. A continuación detallamos mediante cuadros sinópticos, el por qué nuestro aplicativo es factible.

### 2.3.1 Factibilidad Técnica

La VPN como tal es una red virtual, es decir, para su implementación técnica es necesario solamente que funcione nuestra red principal correctamente y tener un cliente externo con acceso a internet. Todos los dispositivos que involucran este proyecto están disponibles en el mercado y son los siguientes:

| Dispositivo          | Características   | Precio |
|----------------------|---|--------|
| Router               | D-Link DIR-300 (54 MBPS Inalámbrico) 4 Puertos  | \$ 65  |
| Access Point         | D-Link2100AP (2.4 GHz Wireless 108 Mbps)  | \$ 90  |
| Conector RJ45        | Genérico (1 Paquete 100 plufs)  | \$ 12  |
| Cable UTP Cat 6      | Cajón de 100 mts  | \$ 70  |
| Computadora Personal | Intel Core Duo 1.6 Ghz, 1 Gb de Ram, DVD Writer, Disco Duro 160 GB Samsung, Monitor Pantalla Plana LG | \$ 700 |

**Tabla 3** - Precios de varios dispositivos de red y computador

Se tomó como muestra algunos dispositivos a la venta en el mercado y con precios accesibles; para corroborar que existe todo lo necesario para la implementación de una pequeña red de oficina. Los precios de seguro cambiarán con el tiempo.

### 2.3.2 Factibilidad Económica

Los gastos a nivel de software en lo que refiere al sistema operativo en el servidor Linux, el Servidor Apache, el lenguaje PHP, Base de datos y Protocolos a implementar, son de costo cero. Así que detallamos los costos concernientes al resto de recursos que necesitaremos para nuestro aplicativo.

| RECURSO   | Precio Unit. | TOTAL              |
|---|--------------|--------------------|
| Costo de Servicio de Internet por 3 meses                           | \$ 45,00     | \$ 135,00          |
| Costo de Router D-Link Wireless (4 puertos)                         | \$ 65,00     | \$ 65,00           |
| 3 Tarjetas Adicionales de Red                                       | \$ 10,00     | \$ 30,00           |
| Gastos Generales (Agua, Luz, Teléfono) por 2 meses                  | \$ 25,00     | \$ 50,00           |
| 3 Pc's (Intel P4, Mainboard Intel, 512 MB, DVD Writer, Monitor 17") | \$ 500,00    | \$ 1.500,00        |
| 10 Metros de Cable UTP Categoría 6                                  | \$ 0,50      | \$ 5,00            |
| 6 Conectores RJ45   | \$ 0,11      | \$ 0,66            |
| 3 Licencias Windows XP SP2  | \$ 140,00    | \$ 420,00          |
| <b>TOTAL</b>  |              | <b>\$ 2.205,66</b> |

**Tabla 4 - Costos para la realización del aplicativo**

### 2.3.3 Factibilidad Operacional

Nuestro aplicativo va dirigido a las pequeñas y medianas empresas, ya que utilizamos herramientas Open Source, es

decir, nos evitaremos el pagar licencias costosas como la de un Servidor Microsoft o una base de datos Oracle.

Después de hacer un pequeño muestreo en la ciudad, podemos considerar que la mayoría de las empresas, manejan el personal técnico informático de la siguiente forma:

### **De 2 a 15 computadores**

La empresa cuenta con una persona externa, que normalmente realiza cada mes un mantenimiento de computadoras y redes. Esta persona puede tener un título o sólo los conocimientos necesarios para realizar este mantenimiento. Dependiendo de la eficacia de la persona y de sus costos profesionales, puede este técnico durar mucho tiempo dando asesoría a la empresa.

### **De 16 a 50 computadores**

La empresa cuenta con al menos una persona (con título en Sistemas), que se haga cargo del mantenimiento de las computadoras y del buen funcionamiento de la red. Dependiendo de la empresa, en esta categoría existen ya departamentos de Sistemas con un Jefe y al menos 2 ayudantes de Sistema.

### **De 51 a 100 computadores**

La empresa cuenta con un departamento de Sistemas, en donde se maneja la parte de Software y Hardware. Normalmente cuentan con sus propios técnicos y desarrolladores (en caso de que implementen su propio sistema)

### **Conclusión**

Nuestro aplicativo abarcaría más el mercado de las 2 primeras categorías, ya que por lo general estas empresas están interesadas en adquirir soluciones informáticas confiables y funcionales a un bajo costo. En la tercera categoría, la competencia sería más grande porque ellos por lo general estarán en la capacidad de adquirir un dispositivo de hardware (por ejemplo Cisco VPN 300), la cual puede levantar un servicio VPN con más configuraciones que la de nuestro aplicativo.

## **2.4 Diagrama de Flujo de Datos**

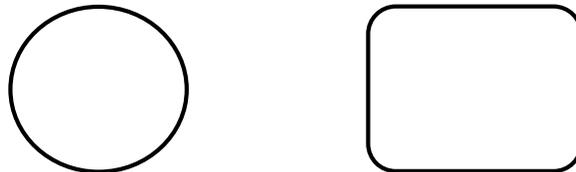
Es un modelo lógico-gráfico para representar el funcionamiento de un sistema en un proyecto software.

Los diagramas derivados de los procesos principales se clasifican en niveles, los cuales son:

- ✓ Nivel 0: Diagrama de contexto.
- ✓ Nivel 1: Diagrama de nivel superior.
- ✓ Nivel 2: Diagrama de detalle o expansión.

Sus elementos gráficos son:

Proceso.- Muestra una parte del sistema que transforma entradas en salidas



**Flujo.-** Se usa para describir el movimiento de bloques o paquetes de información de una parte del sistema a otra.



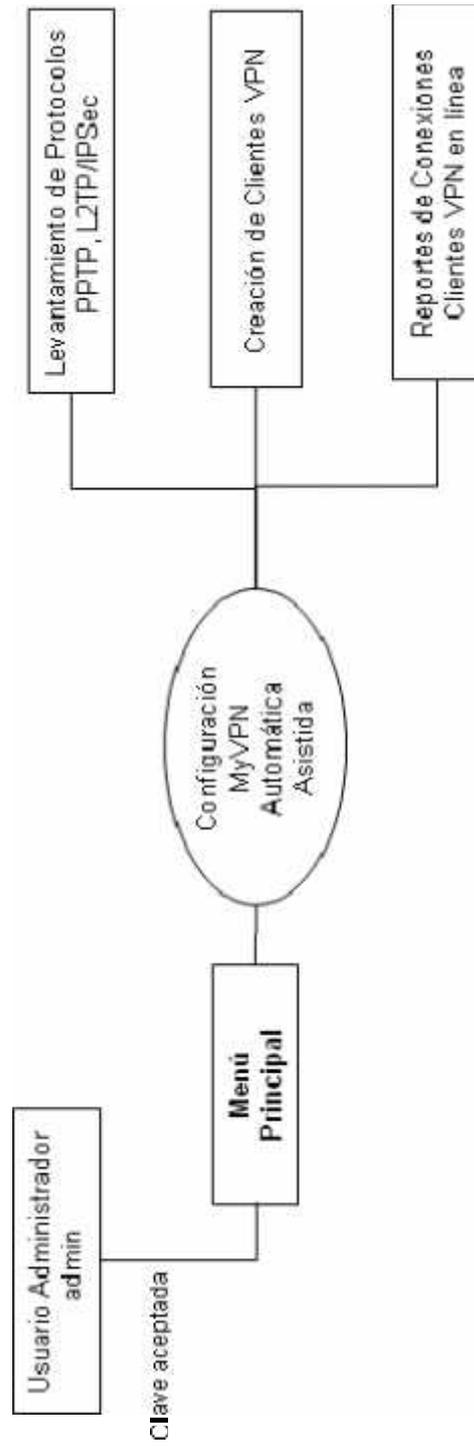
Almacén.- Se utiliza para los paquetes que entran y salen del almacén por medio de flujos.



Terminador.- representan entidades externas con las cuales el sistema se comunica.



**Diagrama de Flujo de Datos  
Nivel 0**



**Gráfico 4a** - Flujo de Datos de MyVPN Nivel 0

### Diagrama de Flujo de Datos de MyVPN Servidor

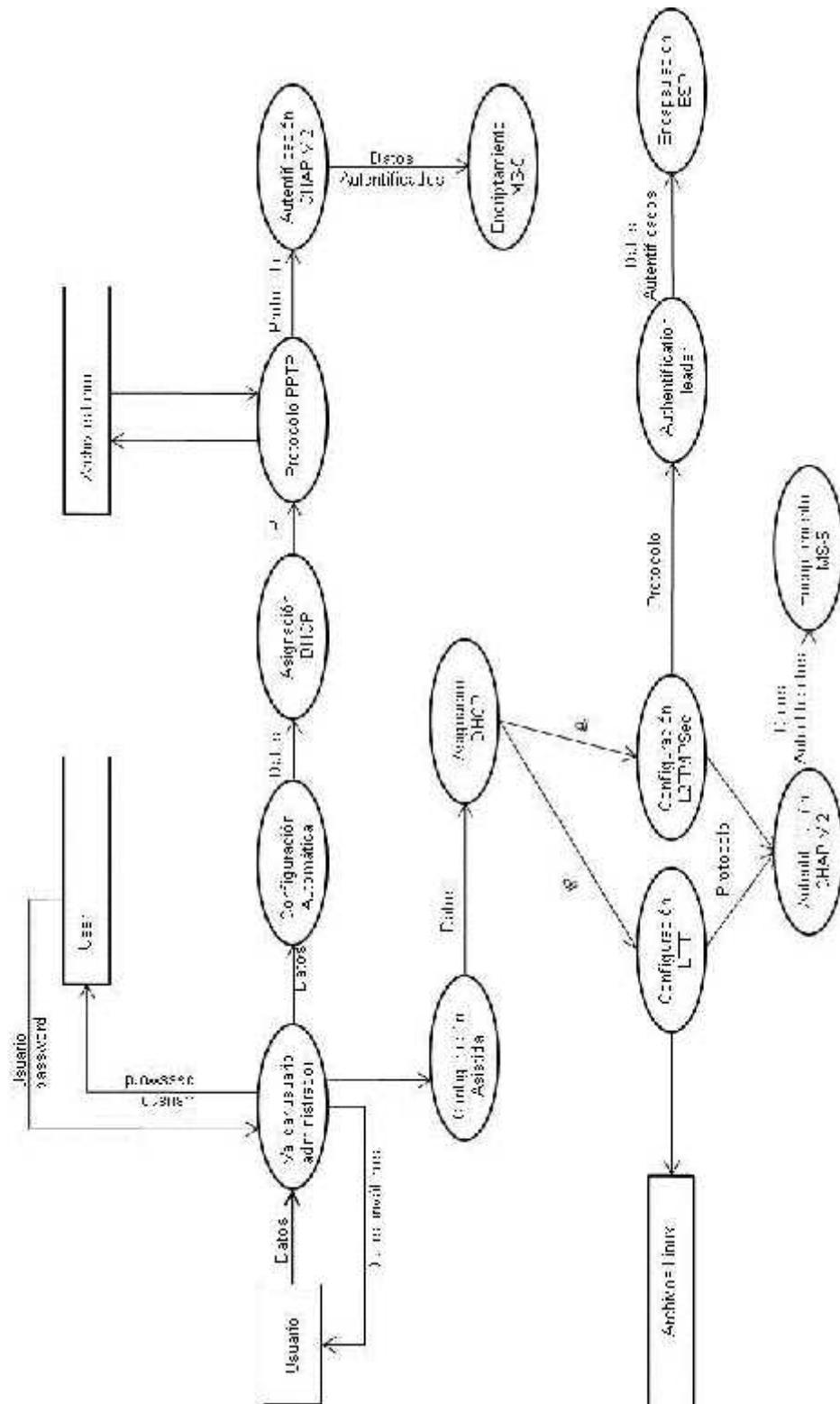


Gráfico 4b - Flujo de Datos de MyVPN

## Diagrama de Flujo de Datos Configuración PPTP

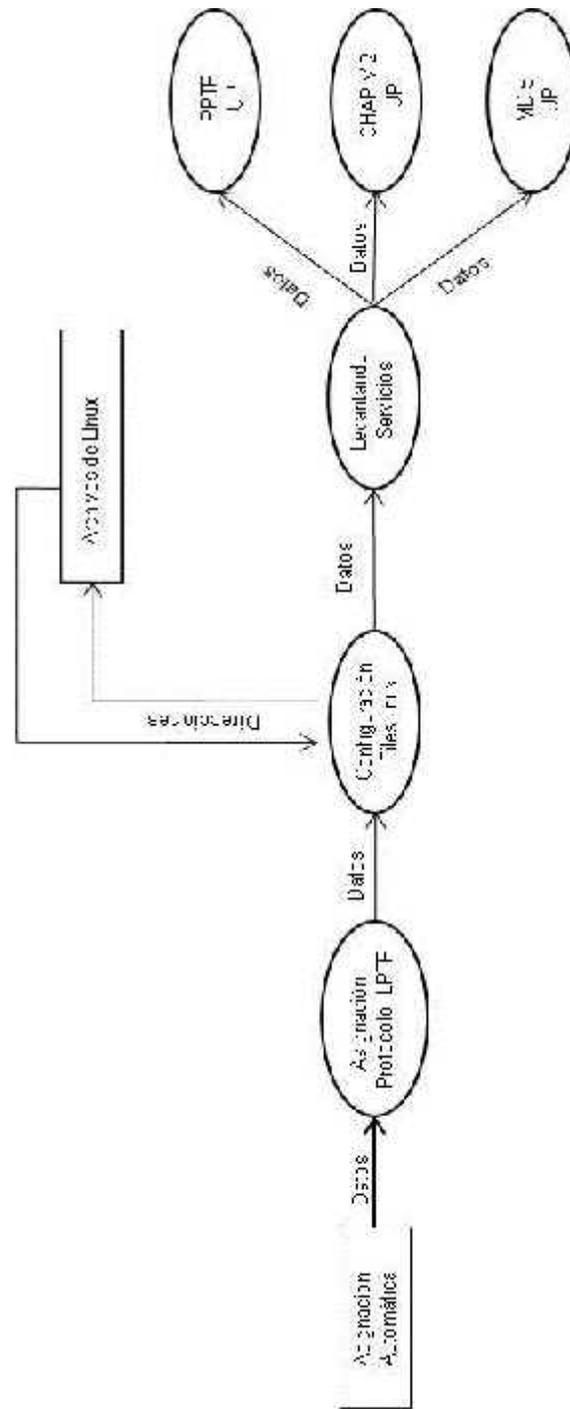


Grafico 4c - Configuración de PPTP

### Diagrama de Flujo de Datos Configuración L2TP / IPsec

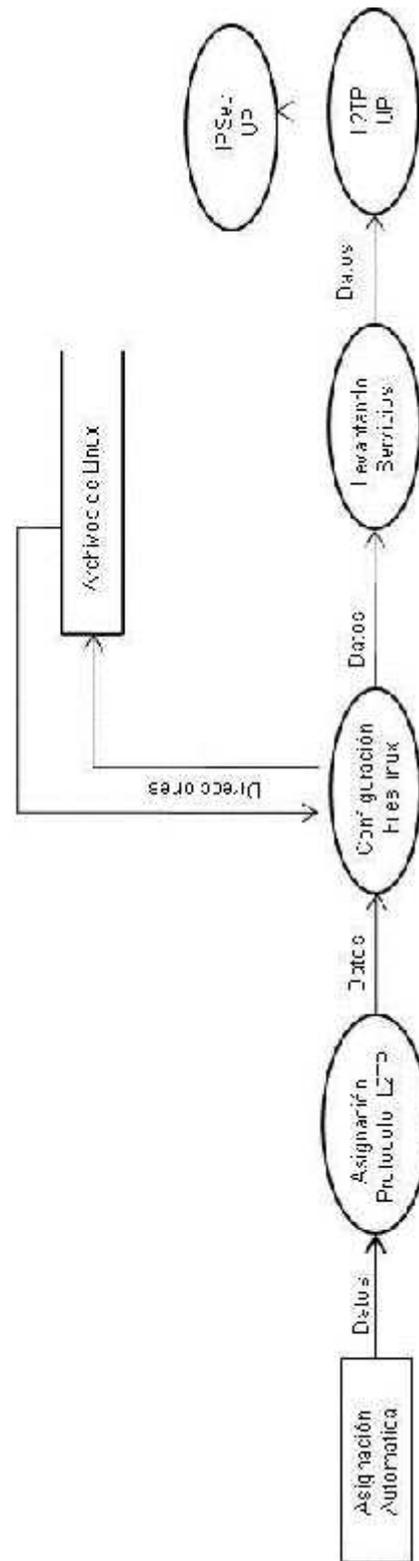
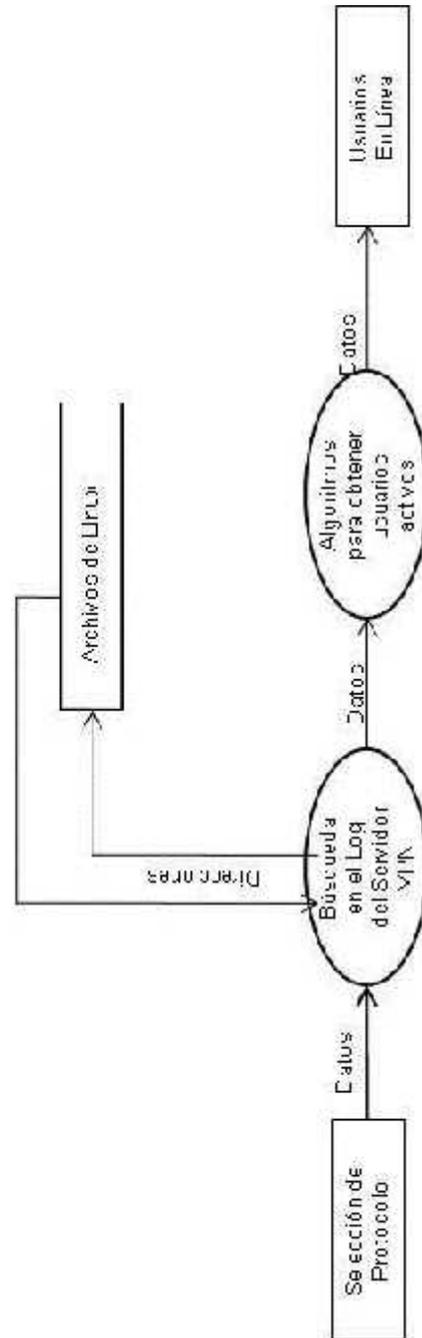


Gráfico 4d - Configuración de L2TP/IPsec

## Diagrama de Flujo de Datos Usuarios en Línea

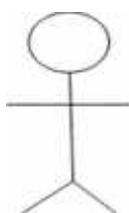


**Grafico 4e** - Estructura de Usuarios en Línea

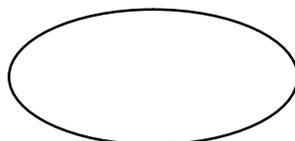
## 2.5 Diagrama de Casos de Uso

Para tener idea de lo que se desea mostrar en nuestro diagrama a continuación se muestra una introducción:

**Actor.-** Representa un tipo de usuario del sistema; en los diagramas de casos de uso los actores se dibujan como una silueta humana.



**Caso de uso.-** Es una tarea que debe poder llevarse a cabo con el apoyo del sistema que se está desarrollando. Se representa con un óvalo.



**Comunicación.-** Este elemento representa la relación que existe entre un Uso-Caso y un Actor, dicho elemento es representado simplemente por una línea recta que se extiende de la figura del actor hacia el ovalo del uso-caso.



### Casos de Uso – Administrador

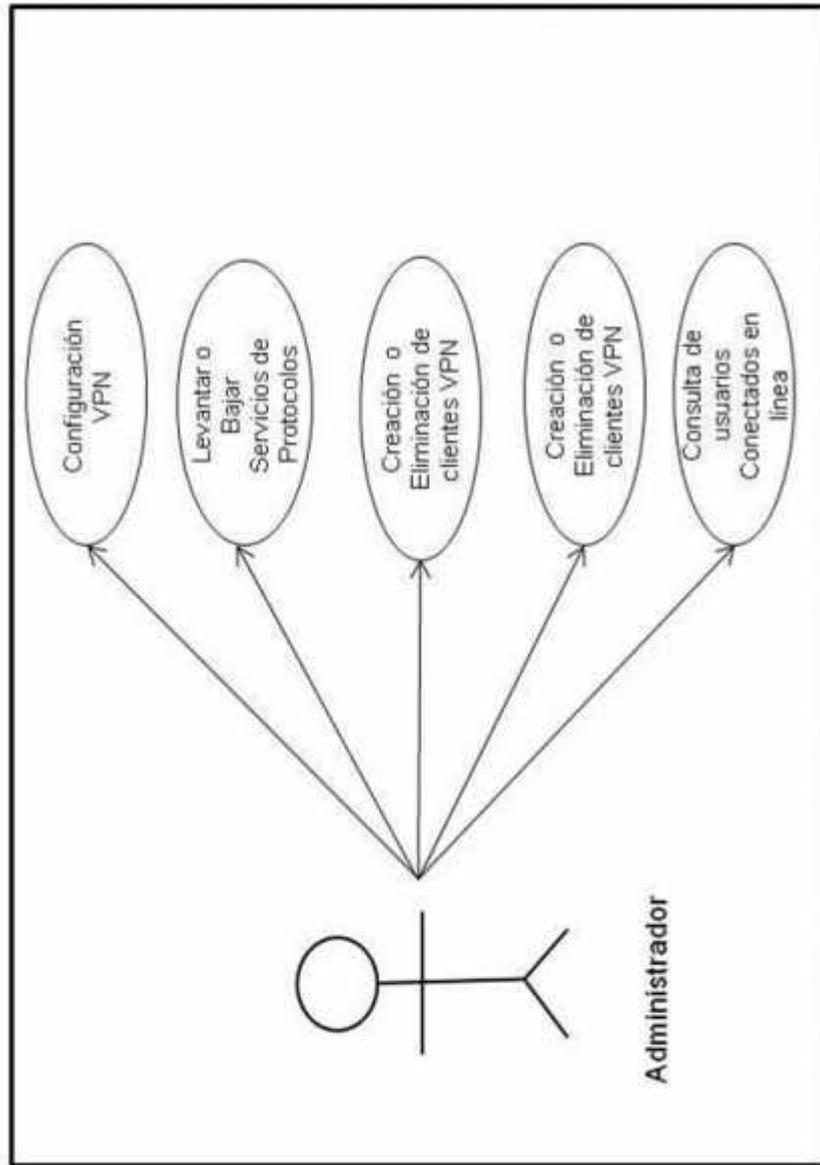


Gráfico 5a - Caso de Uso (Administrador)

### Casos de Uso – Usuario PPTP

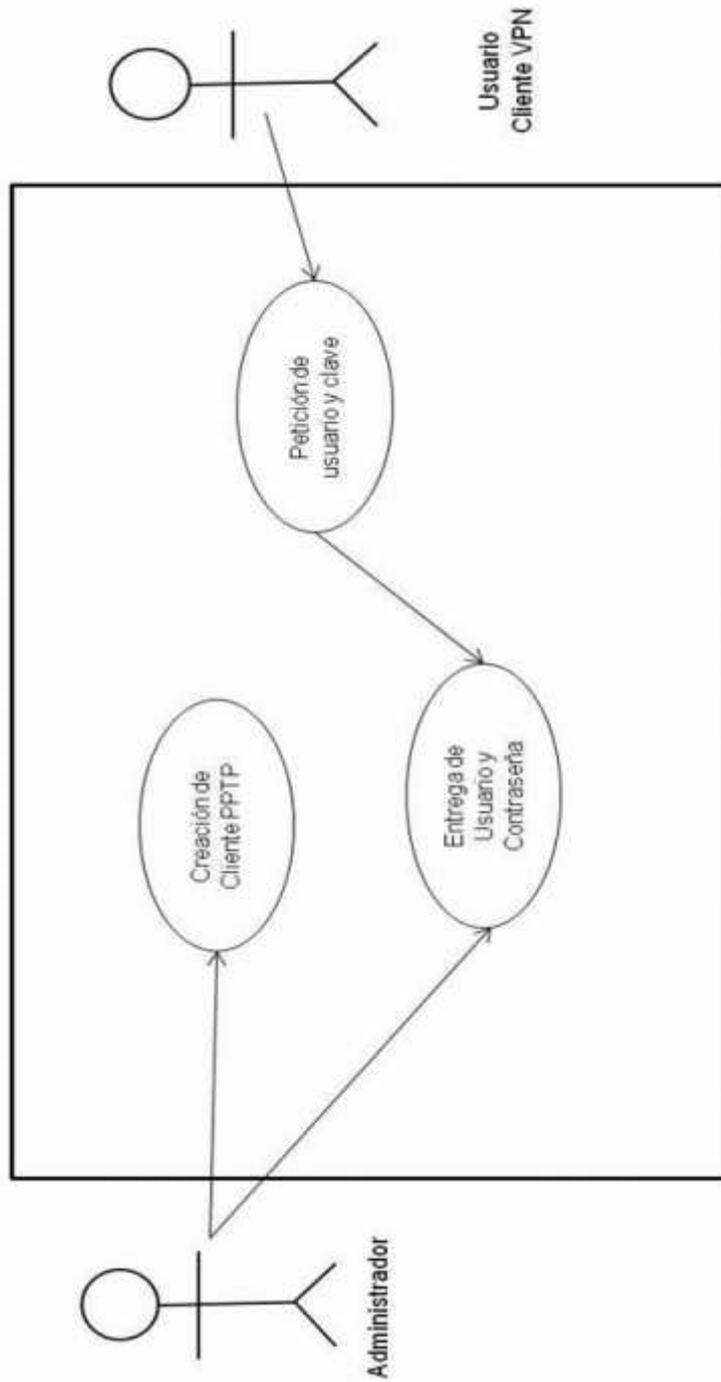


Gráfico 5b - Caso de Uso (Clientes PPTP)

## Casos de Uso – Usuario L2TP/IPSec

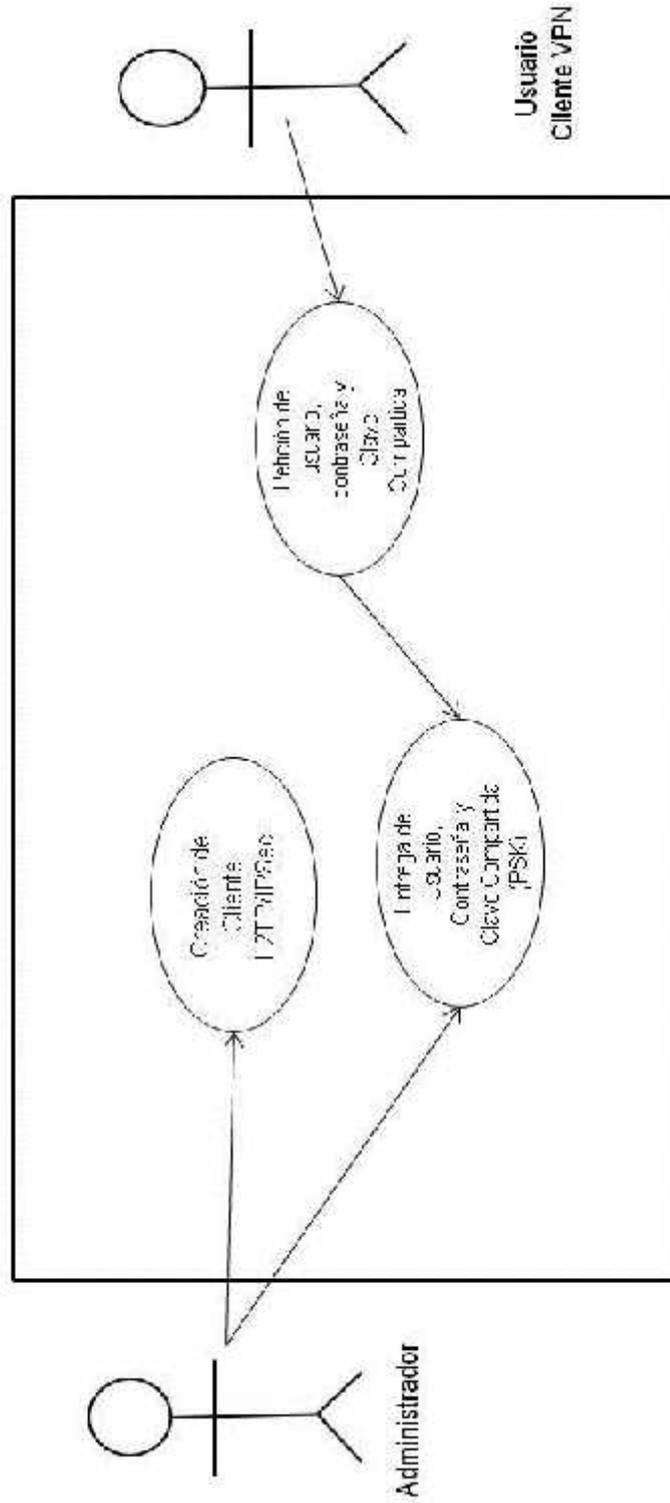


Gráfico 5c - Caso de Uso (Clientes L2TP/IPSec)

# **CAPITULO 3**

## **DISEÑO**

Nuestro aplicativo consta de 2 partes esenciales que son:

### **3.1 Servidor**

A continuación detallaremos los elementos más importantes para la configuración de nuestra VPN en el Servidor Linux.

### 3.1.1 Ingreso

Al ejecutar nuestro aplicativo nos mostrará una pantalla de ingreso, donde deberemos digitar el usuario y password de administrador, la cual puede ser modificada si se desea, la cual tendrá el administrador de redes.

### 3.1.2 Ventana ingreso

**Configuración de VPN**

**INICIO**

### Introducción

VPN (Virtual Private Network): Red privada virtual. Las redes privadas virtuales crean un túnel o conector de un sitio a otro para transferir datos y a ello se le llama encapsulación. Los paquetes de datos van encriptados de forma tal que los datos son ilegibles para los extraños. La VPN debe ser capaz de verificar la identidad de los usuarios y restringir el acceso a la VPN a aquellos usuarios que no están autorizados. Asimismo, debe proporcionar registros estadísticos que muestren quien tuvo acceso, a qué información y cuándo.

### Protocolo PPTP

PPTP (Point-to-Point Tunneling Protocol), es un protocolo desarrollado por Microsoft (MS), 3Com, 3Com/Primary Access, ECI Telematics como caso de uso de protocolo de túnel PPTP Forum, para implementar redes privadas virtuales o VPN.

### Protocolo L2TP/IPSec

L2TP (Layer 2 Tunneling Protocol), da acceso remoto a los clientes a través de una VPN corporativa sobre la red pública con una conexión segura gracias a IPSec (OpenSwan).

**INICIO DE SESIÓN**

Usuario:

Clave:

**Iniciar**

**INFORMACIÓN IMPORTANTE**

Recordemos que este sistema está abierto para un usuario administrador.

© 2008 P-02 | VINGMAIL.COM | HTML 3.1 | Us2 | DESARROLLO POR SOURCE

**Gráfico 6 - Ventana de Ingreso**

También se muestra una pequeña introducción de conceptos básicos sobre VPN, y los protocolos usados en nuestro aplicativo (PPTP y L2TP/IPSEC).

Se podrá ver el estado de la red una vez ingresada el usuario y clave.

Si el usuario y password no son válidos no se tendrá acceso, de ser válido comenzaremos con la creación de nuestra VPN.

### 3.1.3 Configuración de VPN



**Gráfico 7 - Ventana de Configuración**

Dentro de esta pantalla nos disponemos a la creación de nuestra VPN tanto de manera manual (paso a paso) como automática. Podemos ver un botón de Mantenimiento, Estado de Red, Clave de Administrador y los protocolos a usar en la creación de VPN.

Además tenemos un botón de salir, el cual nos direcciona a la pantalla principal

### 3.1.4 Tipo de configuración



**Gráfico 8 - Tipo de Configuración**

En esta pantalla mostramos la configuración manual (paso a paso), dependiendo nuestro grado de conocimiento de redes virtuales escogeremos la configuración apropiada, teniendo como opción la configuración automática (**recomendada**) y la configuración manual ambas válidas para la creación de VPN.

### 3.1.5 Configuración automática



**Gráfico 9** - Ventana de previa Configuración Automática

Nos damos cuenta que esta es una pantalla de bienvenida con una pequeña introducción para no equivocarnos.

Dentro de información importante ubicada en el lado derecho de nuestra pantalla se nos da una pequeña nota.

### 3.1.6 Asignación de IP



The screenshot displays a web-based configuration interface for VPN settings. At the top, there is a dark header with the text "Configuración de VPN" in orange. Below the header, the main content area is white and contains the following elements:

- Sub-header: "Configuración Automática de VPN Usando PPTP"
- Section title: "Ip de Interfases"
- Form fields for IP configuration:
  - "Ip de servidor" with the value "192.168.50.2"
  - "IP Interno" with the value "192.168.20.1"
  - "Rango" with the value "192.168.20.100-110"
- A "Siguiente" button at the bottom of the form.
- A "Cancelar" link to the right of the form.

On the right side of the interface, there is a vertical sidebar with a dark background and orange highlights. It contains the following text:

- "CONFIGURACION INTERNA"
- "Configuración Automática"
- "INFORMACION IMPORTANTE"
- A note: "El presente es un sistema beta elaborado para el usuario básico en redes"

At the bottom of the page, there is a small copyright notice: "COPYRIGHT © 2000 INECS - VPN@INTEL.COM - WITH ALL RIGHTS RESERVED FOR SOURCE".

**Gráfico 10** - Ventana de IP asignadas por Default

Por medio de esta pantalla nos damos cuenta de las IP asignadas automáticamente como la IP del servidor, la IP interna y el rango asignado automáticamente.

No olvidemos el botón salir ubicado en la esquina superior derecha el cual truncara la configuración y nos enviara a la pantalla de inicio.

### 3.1.7 Creación de Usuarios



The screenshot shows a web-based configuration interface for VPN. At the top, there is a dark header with the text "Configuración de VPN" in orange and a "SALIR" button in the top right corner. Below the header, the main content area is white and contains the following elements:

- Text: "Configuración Automática de VPN usando PPTP:"
- Section Title: "Creación de Usuario" in bold black text.
- Form fields:
  - "Usuario:" with an input box.
  - "Contraseña:" with an input box.
  - "Repetir contraseña:" with an input box.
  - "Guardar" button.
- Navigation: "Anterior" and "Siguiente" buttons with arrows, and a "Cancelar" button.

On the right side of the interface, there is a dark sidebar with two sections:

- "CONFIGURACIÓN INTERNA" with a sub-link "Configuración Automática".
- "INFORMACIÓN IMPORTANTE" with a warning message: "Recomendamos que este sistema esté elaborado para un usuario básico en redes".

At the bottom of the screen, there is a footer with copyright information: "COPYRIGHT © 2008 INEPOS | WENDEMAIL.COM | 24741 | 1.088 | 7182450 FOR IPSEC".

**Gráfico 11** - Ventana de Creación de Usuarios

Existen dos criterios para la creación de usuarios que son:

- Creación de usuario dentro de la configuración automática.
- Creación de usuario en mantenimiento de usuario.

#### **3.1.7.1 Creación de usuario dentro de la configuración automática.**

Este tipo de creación es solo dentro de la configuración automática y solo se crearan usuarios PPTP.

#### **3.1.7.2 Creación de usuario en Mantenimiento de Usuario**

Este tipo de creación es el principal, porque es donde creamos usuarios para los dos protocolos.

### **3.1.8 Mantenimiento**

Dentro de esta pantalla se nos mostraran las siguientes opciones:

- Opciones del servidor

Aquí se configurará las IP de WAN y LAN y rango de DHCP en el protocolo que nos encontremos.

- Cuentas de usuario.

Se podrán añadir nuevos usuarios, los cuales pertenecerán al protocolo, que estemos configurando.



**Gráfico 12 - Ventana de Mantenimiento**

- Conexiones activas.  
Muestra los usuarios VPN conectados en el momento
- Conexiones del día.  
Muestra el LOG de usuarios VPN que se conectaron y desconectaron en el transcurso del día.



The screenshot shows a web interface for VPN configuration. The main heading is "Configuración de VPN". The navigation menu includes "INICIO", "PASO A PASO", and "MANTENIMIENTO". The "MANTENIMIENTO" section is active, showing sub-menus for "CONFIGURACION INICIAL", "PROTOCOLOS", and "INFORMACION IMPORTANTE". The main content area is titled "Cuentas de usuario creadas" and contains a table of users.

**Cuentas de usuario creadas**

Las cuentas PPP listadas en esta página se extraen del fichero /etc/ppp/chap-secrets, que se utiliza para la autenticación CHAP. Sólo se muestran las de tu servidor pppd, no las cuentas para hacer conexiones hacia el exterior.

[Nuevo Usuario](#)

| Usuarios | Acción                   |
|----------|--------------------------|
| marco    | <a href="#">Eliminar</a> |
| marco    | <a href="#">Eliminar</a> |
| xavier   | <a href="#">Eliminar</a> |
| xavier   | <a href="#">Eliminar</a> |
| admin    | <a href="#">Eliminar</a> |
| meche    | <a href="#">Eliminar</a> |
| meche    | <a href="#">Eliminar</a> |
| meche    | <a href="#">Eliminar</a> |

[Anterior](#) [Cancelar](#)

Copyright © 2008 CIPRO | VPN@GMAIL.COM | XHTML 1.1 | CSS | Diseñado por BRUNDO

**Gráfico 14 - Cuentas de usuario**

### 3.1.11 Conexiones activas

Nos muestra los usuarios conectados en línea dependiendo del protocolo en el que se haya levantado, los cuales son presentados en una tabla con campos como Fecha, Hora, Protocolo, Usuario, IP.

Como son reportes tomados de un Log de sucesos hay que realizar actualizaciones por eso se cuenta con un botón de *Refrescar*.



**Gráfico 15 - Cuentas activas**

### 3.1.12 Conexiones del día

Esta Pantalla de “Historial de conexiones” muestra los usuarios VPN que se han conectado en el transcurso del día, especificando la hora, el usuario y con qué protocolo se conectaron; así también el momento en que se desconectaron con los datos mencionados en el primer caso. Existe esta pantalla por cada protocolo que implementamos en el sistema.

**Configuración de VPN**

INICIO PRIMERO PASO **MANTENIMIENTO**

### Historial de conexiones

**Conectados**

| Fecha | Hora     | Protocolo | Usuario | IP            |
|-------|----------|-----------|---------|---------------|
| 2014  | 07:08:16 | pptpd     | max     | 192.168.30.58 |
| 2014  | 07:56:43 | pptpd     | max     | 192.168.30.58 |
| 2014  | 08:09:02 | pptpd     | max     | 192.168.30.58 |

**Desconectados**

| Fecha | Hora     | Protocolo | Usuario | IP |
|-------|----------|-----------|---------|----|
| 2014  | 07:54:58 | pptpd     | max     |    |
| 2014  | 07:57:19 | pptpd     | max     |    |
| 2014  | 08:48:17 | pptpd     | max     |    |

[Refrasear](#) [Cancelar](#)

**CONFIGURACION INTERNA**  
 Configuración Automática  
 Estado de Pac  
 Clave Administrador

**PROTOCOLOS**  
 L2TP/IPSEC

**INFORMACION IMPORTANTE**  
 Recordemos que este sistema está diseñado para funcionar bajo Windows

**Gráfico 16 - Conexiones del día**

Además aquí el Administrador tendrá la opción de guardar el log de conexiones en el día, a la Base de Datos (Dando clic en el botón Guardar), para poder realizar algún tipo de auditoría en el momento que él requiera.

## **3.2 Cliente**

Para comenzar la ejecución de nuestro aplicativo sobre los clientes Windows se deberá de configurar la tarjeta de red y dependiendo del sistema operativo archivos de red como Host, DNS, IP, mascara de red entre otros.

### **3.2.1 Ingreso**

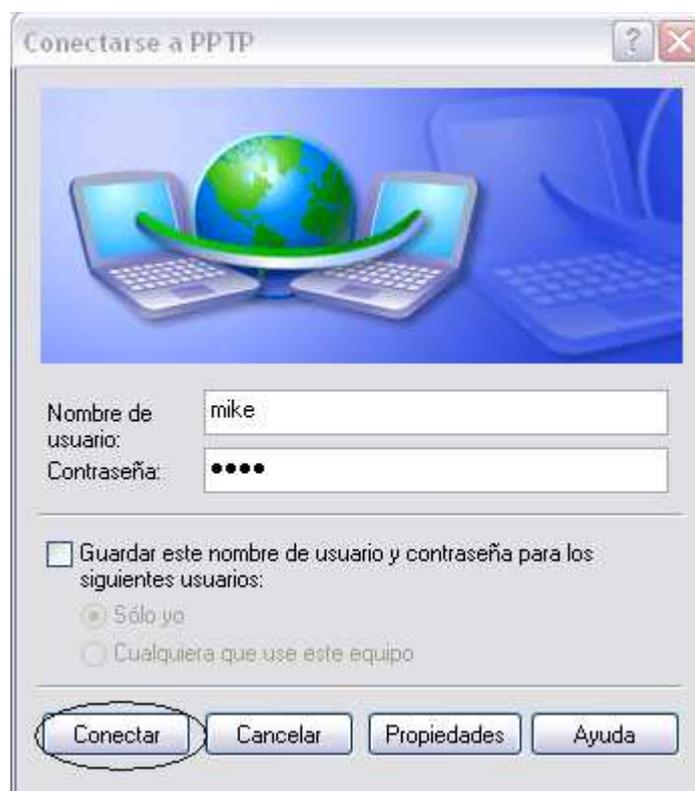
En esta parte el aplicativo se ejecutará dentro del cliente (Windows) para poder conectarse a la red virtual VPN.

Previamente deberá recibir un nombre de usuario y contraseña dado por el administrador de la VPN, quien contará con una IP de VPN. Para la configuración de un cliente Windows nos ayudaremos del Asistente de Creación de Conexiones de Windows, el cual se encuentra en la siguiente ruta:

1. Conexiones de red.
2. Crear una conexión nueva.
3. Conectarse a la red de mi lugar de trabajo.
4. Conexión de red virtual privada.

Y la información que se pide a continuación será obtenida del administrador de redes.

### 3.2.2 Ventana de Conexión para Clientes VPN



**Gráfico 17** - Ventana de Ingreso de Usuario y Contraseña en Clientes Windows

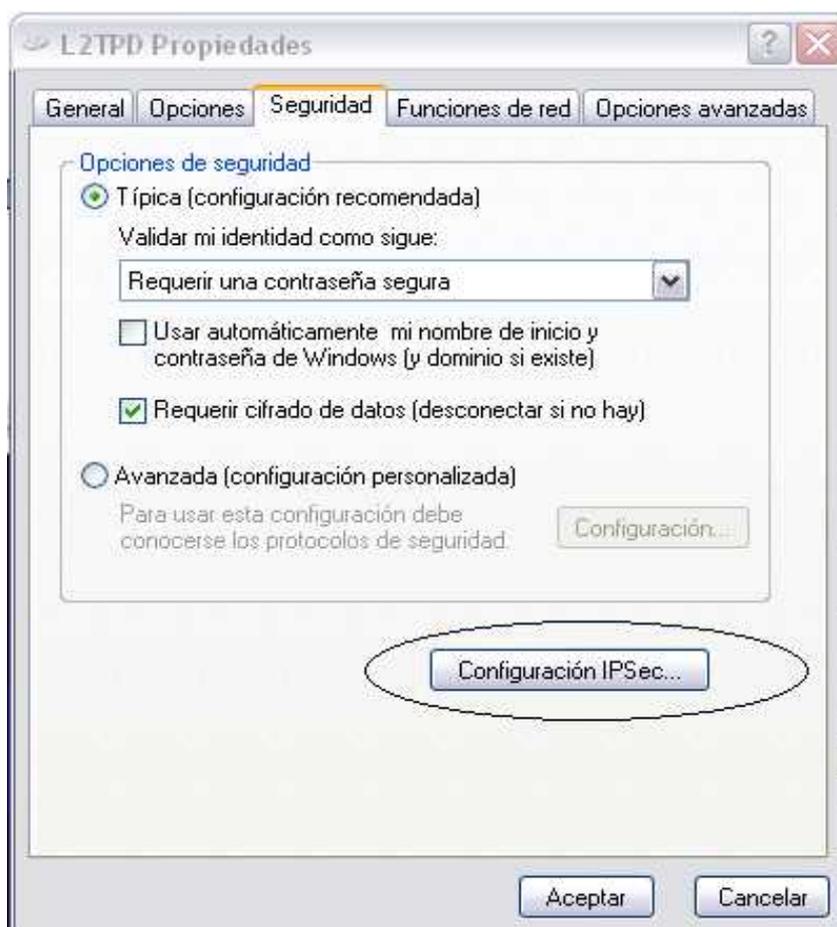
Si vamos a ingresar un usuario PPTP, este ingreso será directo, solo necesitaremos el usuario y password previamente obtenido del administrador de red.

### 3.2.3 Ajustes adicionales para clientes L2TP/IPSec

Si vamos a ingresar como usuario L2tp/IPSec deberemos primero configurar dentro de propiedades de la Ventana de Conexión los siguientes parámetros:

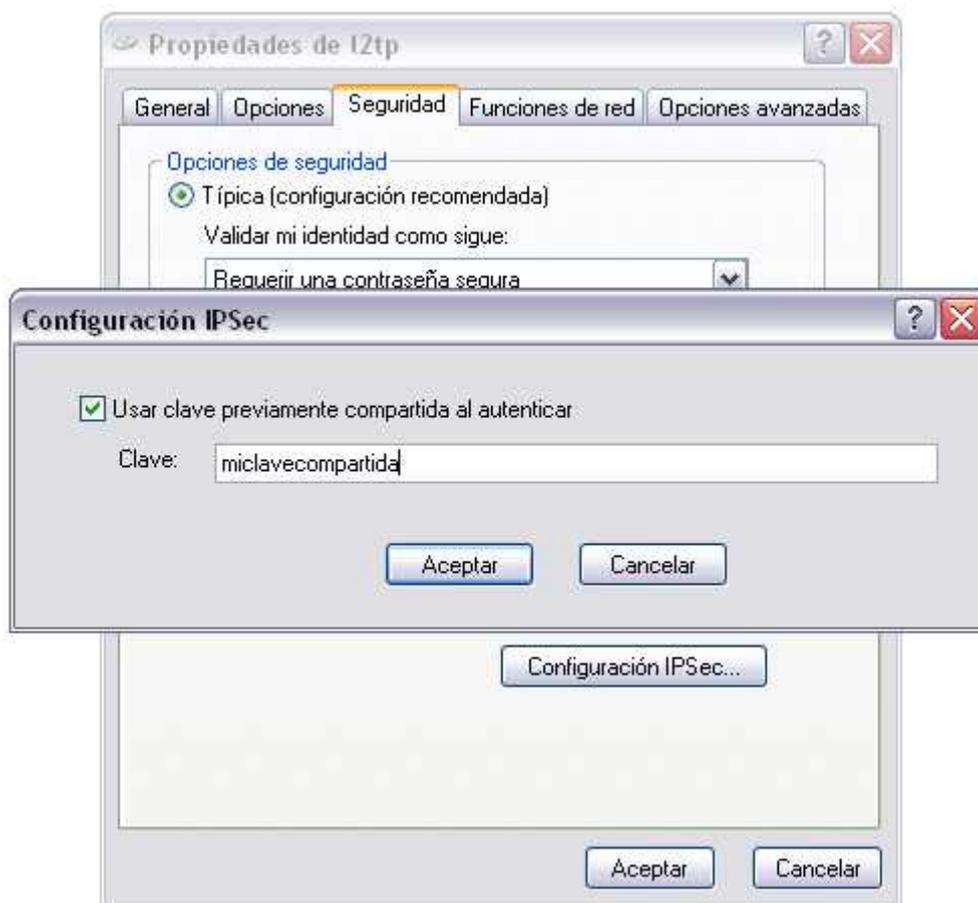
### 3.2.3.1 Seguridad

Donde daremos clic en el botón **Configuración IPsec** en el cual se nos solicitara una clave la cual se nos dio en el momento que se configuró este protocolo en el Servidor VPN.



**Gráfico 18** - Pestaña de Seguridad en Propiedades de Conexión

Al momento se nos mostrará la siguiente pantalla en donde introduciremos nuestra **Clave compartida** la misma que configuramos en nuestro protocolo L2TP/IPsec.

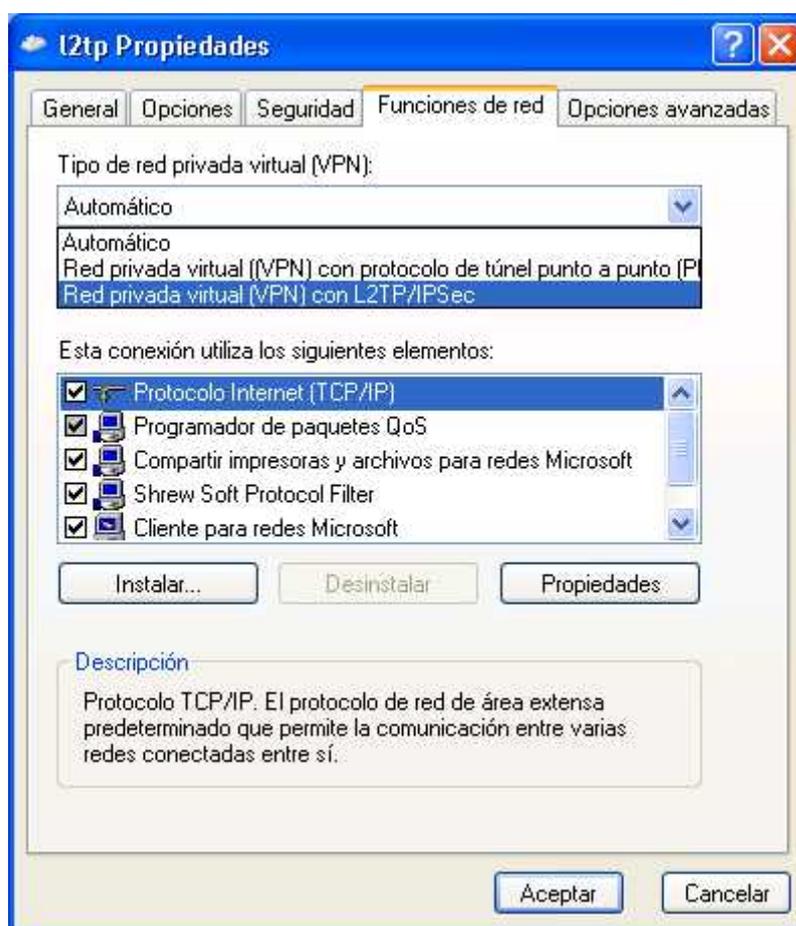


**Gráfico 19 - Ingreso de Clave Compartida**

Esta clave compartida es de uso exclusivo para los clientes VPN y el Administrador deberá procurar que otro usuario ajeno pueda obtener o ver esta clave, para efectos de seguridad

### **3.2.3.2 Funciones de red**

Ahora damos clic en la pestaña funciones de Red y escoger en Tipo de Red Privada Virtual, la opción VPN con L2TP/IPSec.



**Gráfico 20** - Selección de VPN con L2TP/IPSec en Propiedades de Conexión

Esta configuración es solo para L2TP/IPSec y en segundo recuadro se deja la opción que viene por default Protocolo Internet (TCP/IP), damos clic sobre aceptar y continuamos.

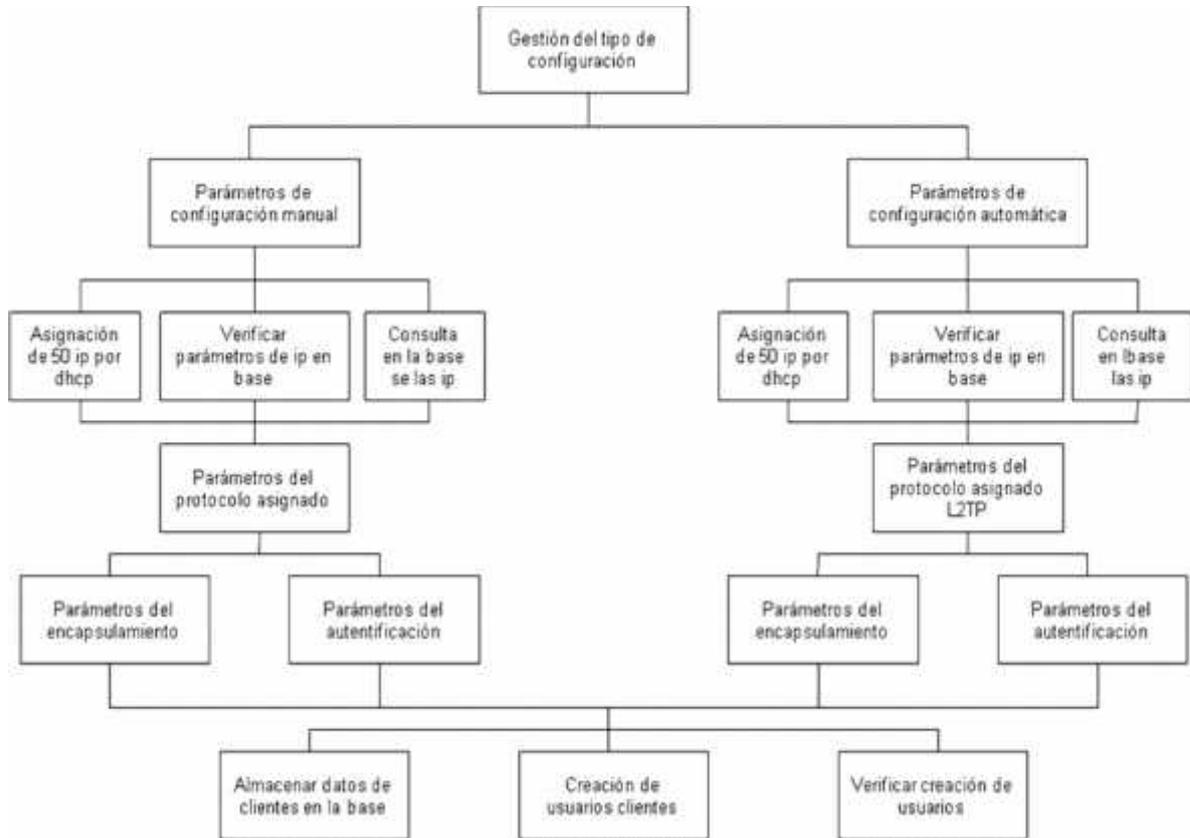
### 3.3 Diseño arquitectónico



**Grafico 21a** - Diseño arquitectónico (Validación Usuario)



**Grafico 21b** - Diseño arquitectónico (Asignación IP)



**Grafico 21c - Diseño arquitectónico (Gestión de Configuración)**

# Capitulo 4

## DESARROLLO Y PRUEBA DEL SISTEMA

### 4.1. Desarrollo del sistema

Para realizar el desarrollo del software MyVpn elegimos utilizar el lenguaje de programación php del cual nos ayudaremos para la manipulación de archivos de configuración haciendo que esto sea fácil y correcto.

Además php es un lenguaje muy parecido C++, por lo que se nos hizo sencillo la elaboración de funciones, el manejo de arreglos con la manipulación de archivos,

También fue necesario e importante un comando del Sistema Operativo Linux **SUDO**.

El comando **SUDO**, a través del cual podemos ejecutar otros comandos con permisos de súper usuario, muy necesario para la comunicación entre php y Linux.

#### 4.1.1. Creación de base de datos

El software MyVpn consta de una base de datos creada bajo PostgreSQL versión 8.1, esta base forma parte de Linux ofreciéndonos ventajas como la actualización de la versión en línea y la manipulación directa sin tener que usar algún otro programa interprete.

Contiene 2 tablas, las cuales estarán relacionadas directamente con la administración de los usuarios.

La estructura de las tablas son las siguientes

#### Users

| <b>CAMPO</b>        | <b>TIPO</b>      |
|---------------------|------------------|
| <b>ID_user (pk)</b> | <b>Numero(4)</b> |
| <b>user</b>         | <b>Texto</b>     |
| <b>Pass</b>         | <b>Texto</b>     |

**Tabla 5a - Tabla "Users"**

### Conexiones

| <b>CAMPO</b>     | <b>TIPO</b>  |
|------------------|--------------|
| <b>fecha</b>     | <b>Texto</b> |
| <b>hora</b>      | <b>Texto</b> |
| <b>Protocolo</b> | <b>Texto</b> |
| <b>Usuario</b>   | <b>texto</b> |
| <b>Ip</b>        | <b>texto</b> |
| <b>Estado</b>    | <b>texto</b> |

**Tabla 5b** - Tabla "Conexiones"

#### 4.1.1.1. Descripción de los campos

A continuación vamos a detallar cada uno de los campos de nuestras tablas users y Conexiones.

##### 4.1.1.1.1. Users

###### **ID\_User**

Es un número secuencial y de auto incremento el cual va a servir para la identificación de los usuarios, cabe indicar que este campo es llenado automáticamente por el sistema y es nuestra clave primaria.

**user**

Este campo contiene el nombre con el cual se va a registrar el usuario.

**pass**

Es la clave con la cual el usuario se registrara para tener acceso al sistema, la misma que se encuentra encriptada.

**4.1.1.1.2. Conexiones****Fecha**

En este campo se guarda la fecha obtenida del Log del sistema contiene el mes y día.

**Hora**

Se guardara en este campo la hora minuto y segundo de conexión del usuario.

**Protocolo**

Tiene guardado el protocolo de conexión y desconexión de los usuarios.

**Usuario**

Dentro de este campo se tendrá una referencia del usuario VPN conectado y desconectado.

**IP**

Guardaré la IP que se genere en el Log del sistema cuando sea un usuario PPTP o L2TP.

**Estado**

Aquí se utilizan dos constantes **d** = desconectados y **c** = conectados.

**4.1.2. Creación de los Componentes**

Para la creación de los componentes se utilizó la metodología orientada a objetos, siguiendo el modelo de desarrollo en espiral, es decir, cuando se termina un módulo inmediatamente se inicia el otro, o de ser necesario se los trabaja en paralelo.

El orden de desarrollo se detalla a continuación:

- Autenticación
- Escaneo
- Análisis de datos

- Almacenamiento de datos
- Cambios sugeridos por el usuario.
- Reporte

#### **4.1.2.1. Autenticación**

Dentro de este módulo se validará el ingreso a nuestro software desde la parte del administrador, así como también el mantenimiento de usuarios (creación, modificación y eliminación).

#### **4.1.2.2. Escaneo**

Este proceso se realiza para llevar un control de las redes a usar, así como los archivos que van a ser utilizados para la ejecución de nuestro software.

#### **4.1.2.3. Análisis de Datos**

Es la validación de los tipos de datos a ingresar dentro de nuestro software, ya sea antes y durante la implementación de nuestro software.

#### **4.1.2.4. Almacenamiento de datos**

La base de datos es alimentada con datos básicos como el usuario, administrador y contraseña, además de datos adicionales para la validación del ingreso a los diferentes módulos del

software. También será alimentada dentro del mantenimiento de usuarios.

#### **4.1.2.5. Cambios sugeridos por el usuario**

Como proveedor del sistema nos interesa saber las sugerencias de los usuarios, para de esta forma implementar mejoras y hacer un aplicativo de fácil uso, cumpliendo de esta forma el objetivo de nuestro software “crear una necesidad y satisfacerla eficientemente en su totalidad”.

#### **4.1.2.6. Reporte**

Es un módulo sumamente importante debido a que nos muestra en línea los estatus en los que se encuentran los usuarios:

- Conectado (On line).
- Desconectado (Off line)

El proceso en mención es secuencial hasta al prototipo final que cumple con los objetivos definidos en el proyecto.

#### **4.1.3. Seguridades**

Las seguridades del sistema están enmarcadas en políticas de uso:

- El nombre del usuario será por default “administrador”, el cual no consta de una clave para poder ingresar al sistema.
- El administrador, al momento de ingresar al sistema podrá otorgarle a su usuario una clave por seguridad, sin embargo, cabe recalcar que esto queda a criterio del administrador y no es un requisito establecido.

Los parámetros de seguridad están marcados bajo los siguientes aspectos:

- Sistema operativo
- Base de Datos
- Humano

#### **4.1.3.1. Sistema Operativo**

En cuanto a nuestro sistema operativo podemos decir que se basa en funciones de servidor por lo cual debería ser manejado por un usuario que tenga conocimientos de administración de servidores y redes. Cada sistema operativo maneja sus propios niveles de seguridad, y si no se respetan las condiciones para que el programa pueda trabajar, no podrá ejecutarse nuestro software.

#### **4.1.3.2. Base de Datos**

Dentro de nuestra base de datos será guardada la información de cada usuario que haya sido registrado en nuestro software. Es muy importante que el usuario que administre la base tenga un nivel de conocimiento intermedio, por si llegara a ocurrir algún problema en la ejecución de consultas para generar nuestros reportes.

#### **4.1.3.3. Humano**

Es muy importante destacar que es el parámetro más difícil de controlar debido a que un usuario sin ningún tipo de conocimiento sobre el sistema operativo y redes puede hacer mal uso del software y de esta forma desprestigiar el buen funcionamiento del aplicativo, además de manipular incorrectamente el software perdiendo de esta forma los beneficios que este otorga a la empresa.

### **4.2. Pruebas del Sistema**

Las pruebas del sistema se realizan para verificar el correcto funcionamiento del software y así poder corregirlas antes de entregarlas al usuario final, en este sistema se han realizado diversos tipos de pruebas entre las que más se destacan tenemos:

- Verificación y validación
- Pruebas de unidad
- Pruebas de integración
- Pruebas de seguridad

#### **4.2.1. Verificación y Validación**

Estos términos se complementan, debido a que el tipo de prueba verificación analiza que no existan errores en la implementación del software, y la validación analiza que lo que se define en las especificaciones iniciales guarde relación con lo desarrollado.

Luego de realizar varios tipos de pruebas, en especial con los permisos de los archivos y datos que el usuario podría ingresar al sistema y se corrigieron todos los posibles errores de ingreso que se pudieran presentar al momento de la ejecución con codificación extra en todos los casos.

#### **4.2.2. Pruebas de Unidad**

En este tipo de pruebas, se evalúa el desempeño de cada módulo de manera independiente, y forma parte del tipo de pruebas de la caja blanca, es decir analizan procesos de cada modulo, para probar coherencias.

Se realizó este tipo de prueba en los siguientes módulos:

- Autenticación
- Análisis
- Registro de datos
- Reportes

#### **4.2.2.1. Autenticación**

Se verificaron las clases que me dan acceso al sistema, se comprobó su operatividad, me dan máximo 3 intentos de autenticarme, sino me trunca el acceso.

#### **4.2.2.2. Análisis**

Se establecieron pantallas por función para de esta forma poder verificar el resultado final de la operación, logrando así comprobar el envío de los parámetros correctos como también la devolución del valor esperado.

#### **4.2.2.3. Reporte**

Dentro de las prueba de reportes, buscamos el mejor criterio para armar la presentación de datos requeridos por el administrador de nuestro software.

#### **4.2.2.4. Pruebas de integración**

Se realizan las pruebas necesarias por parte de cada programador, añadiendo criterios sobre mejoras que podrían aportar alguna ayuda al desempeño de sus actividades.

Para complementar las pruebas hemos aplicado las dos técnicas más usadas que son: técnicas de caja blanca a nivel de componentes y la técnica de caja negra a nivel de interfaz.

Este sistema fue evaluado por nuestro catedrático de seminario el Ing. Carlos Montes y administradores de sistemas basados en la plataforma Linux los mismos que aportaron con su criterio y nos aconsejaron algunas modificaciones.

# Capitulo 5

## IMPLEMENTACIÓN DEL SISTEMA

Una vez llegado al prototipo final del sistema MyVpn se procedió a la instalación del software en el equipo especificado, con el fin de presentarlo y sustentarlo ante el tribunal que revisó nuestro proyecto. Recalamos una vez más que el sistema fue desarrollado totalmente bajo ambiente Open Source, debido a los beneficios que esto representa, entre los cuales citamos:

- El costo, como tecnología libre, minimiza los gastos de licencias para la empresa, volviéndolos prácticamente en cero costo.
- El código abierto, es decir, que cualquier persona lo puede analizar, mejorar y publicarlo, para los demás usuarios técnicos interesados, además de crear mejoras, se puede desarrollar versiones mejoradas del mismo en conjunto con muchas personas de la web.
- Multiplataforma, más que nada funciona en diversas plataformas Linux, pero su uso óptimo se dé en maquinas con Centos, el lenguaje PHP permite esta característica, el cual puede ser fácilmente modificado para trabajar, incluso en plataformas Windows sin problema. La base de datos es de tipo open source, este criterio no impide igual que el programa pueda ser usado con cualquier otro tipo de base de datos, realizando el mantenimiento necesario.

## 5.1 Elementos Físicos

El elemento físico que utilizamos para la implementación del sistema fue un computador Clon de las siguientes características:

- Pentium IV 2.4 GHZ
- 1 GB de memoria RAM
- Disco Duro de 80GB de 7200RPM
- CD/DVD Writer
- Monitor de 17"
- Teclado y Mouse
- 2 Tarjetas de Red (Interface WAN y LAN)

Aclaremos que la primera interfaz, la utilizaremos para nuestra red WAN para que tenga acceso a internet por medio de una IP Pública, esto es indispensable para que se pueda establecer el túnel VPN.

La segunda interfaz de red será para la LAN de la compañía, con la que los clientes VPN, se podrán conectar a través de internet. También los clientes VPN necesitarán al menos de una IP Pública para conectarse al Servidor VPN.

## 5.2 Elementos Lógicos

Se deberá implementar el computador principal, con las siguientes herramientas:

- Tener instalado y en operación el sistema operativo Linux distribución CentoS 5.0, correctamente configurado en el cual se va a desarrollar el proyecto. Más adelante se detalla paso a paso la instalación y configuración del mismo para poder empezar a desarrollar el proyecto.
- El sistema operativo, una vez listo para trabajar, deberá tener instalados los siguientes paquetes, los cuales se proporciona en los CD's del proyecto y los de protocolo en páginas de Internet que detallamos en la bibliografía.
  - ✓ httpd-2.2.3-11.e15\_1.centos.3 (Servidor Apache)
  - ✓ php-5.1.6-5.e15 y php-pgsql-5.1.6-5.e15 (PHP V.5)
  - ✓ postgresql-8.1.11-1.e15\_1.1 (Base de datos POSTGRES)
  - ✓ bind-9.3.3-10.e15 (DNS de Linux)
  - ✓ pptpd-1.3.4-1 (Paquete para PPTP)
  - ✓ l2tpd-0.69-12jdl (Paquete para L2TP)
  - ✓ openswan-2.4.9-2.el5.kb (Paquete para IPSec)
  - ✓ sudo-1.6.8p12-10 (Paquete para SUDO)
  - ✓ iptables-1.3.5-1.2.1 (Firewall nativo de Linux)

### **5.3 Elementos Humanos**

Desarrolladores:

- Xavier Guambo Erazo
- David Morante Lombeida
- Miguel Contreras Jordán

Quienes cuentan con sólidos conocimientos en el sistema operativo Linux así como en el lenguaje de programación PHP.

### **5.4 Infraestructura**

La infraestructura básica para que el proyecto se pueda desarrollar sin ningún problema es la siguiente:

- Cableado de red usando como medio de transmisión: cable par trenzado categoría 5e.
- Si existen más de dos maquinas, se necesitará configurar un router, para conectar todas las maquinas. Si ya hay una infraestructura de red, se la usará como medio de transmisión.
- Tarjetas de red de preferencia 3-Com por su compatibilidad con sistemas Linux, tipo 10/100.

# **CAPITULO 6**

## **RECOMENDACIONES Y**

## **CONCLUSIONES DE LA TESIS**

### **6.1. Recomendaciones**

Las recomendaciones que citaremos tanto a nivel de software y de hardware y de Internet, podrían ser un poco costoso implementarlo, sin embargo, esto hará que nuestro sistema funcione de manera eficiente y rápida.

### 6.1.1. Hardware

Ya que es posible que el servidor VPN tenga que estar operativo 24/7, de preferencia recomendamos los siguientes componentes:

- Procesador Corel Duo E2180 2.0
- 2 GB Ram (DDR2)
- Disco Duro SATA de 160 GB
- DVD Writer
- Monitor de 17"
- Teclado y Mouse
- 2 Tarjetas de Red (3Com)

### 6.1.2. Software

Hacemos incapié en que para la página web de nuestro sistema MyVPN, funcione correctamente, deben instalarse todos los paquetes mencionados, además, gracias a que la distribución Centos 5 cuenta con el comando **yum**, el usuario puede actualizar los paquetes mediante este comando, por ejemplo:

```
yum update pptpd-1.3.4
```

Le advertimos que si se instalan nuevos paquetes de los protocolos y estos contienen una configuración totalmente

diferente de la expuesta en el manual técnico, es muy probable que no se pueda realizar una correcta configuración.

## **6.2. Conclusiones**

Existiendo en el mercado tantos dispositivos especializados que pueden implementar una Red Privada Virtual, el Sistema MyVPN es una alternativa válida a las pequeñas y medianas empresas que deseen implementar esta tecnología a un bajo costo. Gracias a que el desarrollo del sistema fue bajo ambiente Linux, tendrá una mayor acogida debido a dos razones: La primera, porque Linux crece cada día en confianza y seguridad en Servidores y segundo, la instalación del sistema en otra distribución sería muy sencillo, sólo el Administrador tendrá que obtener los paquetes necesarios compatibles con la distribución de Linux, colocar la carpeta del sistema en el directorio del demonio APACHE y ejecutar el script de la Base de Datos Postgres.

Deseamos sinceramente que este sistema sea de utilidad para aquellos que buscan la implementación de la VPN o que en su defecto, que estén en completa libertad de reutilizar el código y mejorarlo o adaptarlo a otros protocolos si así lo desearan.

## **BIBLIOGRAFÍA**

### **Libros consultados**

1. Ingeniería de Software - Pressman
2. Building and Integrating Virtual Private Networks with Openswan – Paul Wauters
3. Red Hat Linux Networking and System Administration – Terry Collings

### **Páginas web consultadas**

1. [www.linuxparatodos.net](http://www.linuxparatodos.net)
2. [www.wingloon.com/2007/11/06/pptp-server-installation-in-centos-5/](http://www.wingloon.com/2007/11/06/pptp-server-installation-in-centos-5/)
3. [www.jacco2.dds.nl/networking/freeswan-l2tp.html](http://www.jacco2.dds.nl/networking/freeswan-l2tp.html)
4. [www.pgadmin.org](http://www.pgadmin.org)
5. [www.en.wikipedia.org/wiki/Layer\\_2\\_Tunneling\\_Protocol](http://www.en.wikipedia.org/wiki/Layer_2_Tunneling_Protocol)
6. [www.rpm.pbone.net/](http://www.rpm.pbone.net/)
7. [www.postgres.org](http://www.postgres.org)
8. [www.rpmfind.net](http://www.rpmfind.net)
9. [www.linuxquestions.org](http://www.linuxquestions.org)
10. [www.linux-tip.net](http://www.linux-tip.net)



**UNIVERSIDAD DE GUAYAQUIL**  
**Facultad de Ciencias Matemáticas y Físicas**  
**Carrera de Ingeniería en Sistemas Computacionales**

Configuración de Redes VPN (Redes Privadas Virtuales) y  
Creación de Clientes VPN – MyVPN

**PROYECTO DE GRADO**

**Previo a la obtención del Título de:**

**INGENIERO EN SISTEMAS COMPUTACIONALES**

**Autores:**

Miguel Ángel Contreras Jordán

Eduardo Xavier Guambo Erazo

David Rafael Morante Lombeida

**GUAYAQUIL - ECUADOR**

**Año 2008**

## **AGRADECIMIENTOS**

En primer lugar agradezco a Dios por haberme dado las fuerzas necesarias para seguir con mis estudios y no desmayar, a mis padres por apoyarme incondicionalmente en esta etapa tan importante de mi vida, gracias a todas las personas que me olvido de mencionar pero están en mis pensamientos.

Miguel Ángel Contreras Jordán

## **AGRADECIMIENTOS**

A Dios por regalarme unos padres maravillosos que me han ayudado en todo momento. A mis hermanos: Fernando y Paola, que creyendo en mí, me llenaron de fuerza. A mi tía la Lcda. Rosario Erazo, mi segunda mamá, le debo tanto también. Al P. Fray Julio Landívar y a la Comunidad de Agustinos, que me proveyeron el mejor consejo en mis momentos de zozobra. A todos los profesores que me inculcaron sus conocimientos en mi etapa universitaria y a tantos buenos amigos y amigas que hice en la misma. A todos ellos, mi sincero y eterno agradecimiento.

*Xavier Guambo*

## **AGRADECIMIENTOS**

A Dios sobre todas las cosas, a mi madre, la Ing. Shirley Lombeida, quien luchó conmigo día a día en este camino de esfuerzo y perseverancia; a toda mi familia que siempre está a mi lado para resolver cualquier problema. A la Ing. Giselly Ycaza M. por esos pequeños detalles que terminan siendo los más importantes, a mis amigos, compañeros y profesores los cuales han contribuido en la culminación de mi carrera.

David Rafael Morante Lombeida

## **DEDICATORIAS**

A Dios, a mis padres y a todas aquellas personas que estuvieron a mi lado cuando más las necesite.

Miguel Ángel Contreras Jordán

## **DEDICATORIAS**

Definitivamente este logro va dedicado a Dios, Fuente de Toda Vida y; a mis padres, el Lcdo. Víctor Hugo Guambo y la Ing. Com. Ana María Erazo, quienes soñaron con este momento con tanta ilusión y entusiasmo, incluso más que yo, a ellos les debo todo lo que soy. Amados padres, espero devolverles con la obtención de este título, al menos un poco de toda la alegría y el bienestar que me han procurado desde que vine a este mundo.

*Xavier Guambo*

## **DEDICATORIAS**

A mis abuelos quienes fueron pilares en este logro, a mi novia Mayra Ajón Y., quien fue mi apoyo en todo momento.

David Rafael Morante Lombeida

## TRIBUNAL DE GRADUACIÓN

---

Presidente del Tribunal

---

Primer Vocal

---

Segundo Vocal

---

Secretario

## DECLARACIÓN EXPRESA

“La autoría de la tesis de grado corresponde exclusivamente al suscrito(s), perteneciendo a la Universidad de Guayaquil los derechos que generen la aplicación de la misma”.

(Reglamento de Graduación de la Carrera de Ingeniería en Sistemas Computacionales, Art. 26).

---

Miguel Angel Contreras Jordán  
mike\_contreras\_jordan@hotmail.com

---

Eduardo Xavier Guambo Erazo  
xavierguambo@hotmail.com

---

David Rafael Morante Lombeida  
david\_korn90@hotmail.com

## RESUMEN

El Sistema MyVPN, fue desarrollado enfocándonos en las necesidades de las medianas empresas, específicamente en la implementación de Redes Privadas Virtuales a un bajo costo, todo esto posible gracias a que solo utilizamos herramientas Open Source y de libre distribución en la Web.

Desarrollado en el lenguaje PHP, con un ambiente amigable e intuitivo, la navegación por el sistema se convierte en tarea sencilla, ya que cuenta con Configuración Asistida y Automática, que guiará al usuario a efectuar una correcta configuración de la VPN.

Le recordamos que nuestro sistema dará una IP a los clientes VPN, de la Red Principal de la LAN configurada en su servidor Linux. Los servicios que pudieran implementarse en esta interface LAN, tales como SAMBA, FTP, etc., depende del administrador dicho servidor.

## ÍNDICE GENERAL

|   |          |
|---|----------|
| AGRADECIMIENTOS.....                                | II       |
| DEDICATORIAS.....                                   | V        |
| TRIBUNAL DE GRADUACIÓN.....                         | VIII     |
| DECLARACIÓN EXPRESA.....                            | IX       |
| RESUMEN.....  | X        |
| ÍNDICE GENERAL.....                                 | XI       |
| <br>  |          |
| <b>MANUAL DE USUARIO.....</b>                       | <b>1</b> |
| INGRESO AL SISTEMA.....                             | 2        |
| CONFIGURACIÓN AUTOMÁTICA POR EL PROTOCOLO PPTP..... | 5        |
| PASO A PASO.....                                    | 8        |
| Por el Protocolo PPTP.....                          | 8        |
| Por el Protocolo L2TP/IPSec.....                    | 11       |
| MANTENIMIENTO DE VPN.....                           | 16       |
| Mantenimiento Por PPTP.....                         | 16       |
| Opciones del Servidor.....                          | 17       |
| Cuentas de Usuarios.....                            | 18       |
| Conexiones Activas.....                             | 19       |
| Conexiones en el Día.....                           | 20       |
| Mantenimiento Por L2TP/IPSec.....                   | 21       |
| Opciones del Servidor.....                          | 22       |
| Clave Compartida.....                               | 23       |
| Cuentas de Usuarios.....                            | 24       |

|  |           |
|--|-----------|
| Conexiones Activas.....  | 25        |
| Conexiones en el Día .....   | 26        |
| CONFIGURACIÓN DE LA RED .....  | 27        |
| CAMBIO DE CLAVE DE ADMINISTRADOR .....   | 28        |
| <b>MANUAL TÉCNICO .....</b>  | <b>29</b> |
| INTRODUCCIÓN .....   | 29        |
| CONFIGURACIÓN DEL SERVIDOR .....   | 29        |
| Archivos de Configuración de PPTP .....  | 29        |
| Archivos de Configuración de L2TP/IPSEC.....   | 30        |
| REGLAS DE FIREWALL .....   | 34        |
| PROCEDIMIENTO PARA LA CREACIÓN DE CONEXIÓN DE LOS USUARIOS QUE SE<br>CONECTARÁN A LA VPN ..... | 36        |
| Creación de conexión de PPTP .....   | 36        |
| CONEXIÓN AL SERVIDOR VPN POR PPTP .....  | 40        |
| Creación de conexión de L2TP/IPSEC .....   | 42        |
| CÓDIGO FUENTE PRINCIPAL .....  | 45        |
| Función Config_reg.php .....   | 45        |
| Función DBConnection.php .....   | 46        |
| Función etc.php .....  | 48        |
| Página Index.php.....  | 49        |
| JSON.php .....   | 54        |
| Mante_l2tp.php .....   | 72        |
| Mante_pptpd.php.....   | 74        |
| pptpd.php.....   | 75        |
| Settings.php.....  | 76        |
| Users_pptpd.php .....  | 77        |
| Función Chap-secrets.php.....  | 78        |
| Función Connl2tpd.php.....   | 79        |

|                                   |    |
|-----------------------------------|----|
| Función elimUsrPPP.php.....       | 81 |
| Función LeePtpd.php.....          | 82 |
| Función LeeUsrPPP.php .....       | 83 |
| Función Nuevousrppp.php.....      | 84 |
| Función modificaOptL2tp.php ..... | 85 |
| Función modificaOptPtpd.php.....  | 86 |

## ÍNDICE DE GRÁFICOS

|  |    |
|--|----|
| <b>Gráfico 1</b> – Pantalla de Inicio .....                                    | 3  |
| <b>Gráfico 2</b> – Pantalla de Bienvenida.....                                 | 4  |
| <b>Gráfico 3</b> – Configuración Automática PPTP Paso 1 .....                  | 5  |
| <b>Gráfico 4a</b> – Configuración Automática PPTP Paso 2.....                  | 6  |
| <b>Gráfico 4b</b> – Configuración Automática PPTP Paso 2 (Caso de Error) ..... | 7  |
| <b>Gráfico 5</b> – Configuración Automática PPTP Paso 3.....                   | 8  |
| <b>Gráfico 6a</b> – Paso a Paso por PPTP Paso 1 .....                          | 9  |
| <b>Gráfico 6b</b> – Paso a Paso por PPTP Paso 2 .....                          | 10 |
| <b>Gráfico 7a</b> – Paso a Paso por L2TP/IPSec Paso 1 .....                    | 11 |
| <b>Gráfico 7b</b> – Paso a Paso por L2TP/IPSec (Validación IP) .....           | 12 |
| <b>Gráfico 8</b> – Paso a Paso por L2TP/IPSec – Paso 2.....                    | 13 |
| <b>Gráfico 9</b> – Paso a Paso por L2TP/IPSec – Paso 3.....                    | 14 |
| <b>Gráfico 10</b> – Paso a Paso por L2TP/IPSec – Paso 4.....                   | 15 |
| <b>Gráfico 11</b> – Pantalla Principal Mantenimiento Protocolo PPTP .....      | 16 |
| <b>Gráfico 12</b> – Opciones del Servidor .....                                | 17 |
| <b>Gráfico 13</b> – Cuentas de Usuarios .....                                  | 18 |
| <b>Gráfico 14</b> – Cuentas de Usuarios .....                                  | 19 |
| <b>Gráfico 15</b> – Conexiones en el día.....                                  | 20 |
| <b>Gráfico 16</b> – Mantenimiento por L2TP/IPSec.....                          | 21 |
| <b>Gráfico 17</b> – Opciones del Servidor .....                                | 22 |
| <b>Gráfico 18</b> – Clave Compartida.....                                      | 23 |

|  |    |
|--|----|
| <b>Gráfico 19</b> – Cuentas de Usuarios .....                                  | 24 |
| <b>Gráfico 20</b> – Conexiones Activas .....                                   | 25 |
| <b>Gráfico 21</b> – Conexiones en el día.....                                  | 26 |
| <b>Gráfico 22</b> – Configuración de la Red.....                               | 27 |
| <b>Gráfico 23</b> – Cambio de Clave del Administrador .....                    | 28 |
| <b>Gráfico 24</b> – Archivo de Configuración de L2TP/IPSec .....               | 34 |
| <b>Gráfico 25</b> – Reglas de Firewall para protocolos PPTP y L2TP/IPSec ..... | 35 |
| <b>Gráfico 26</b> – Conexión de usuarios por PPTP – Paso 1 .....               | 36 |
| <b>Gráfico 27</b> – Conexión de usuarios por PPTP – Paso 2 .....               | 37 |
| <b>Gráfico 28</b> – Conexión de usuarios por PPTP – Paso 3 .....               | 37 |
| <b>Gráfico 29</b> – Conexión de usuarios por PPTP – Paso 4 .....               | 38 |
| <b>Gráfico 30</b> – Conexión de usuarios por PPTP – Paso 5 .....               | 38 |
| <b>Gráfico 31</b> – Conexión de usuarios por PPTP – Paso 6 .....               | 39 |
| <b>Gráfico 32</b> – Conexión de usuarios por PPTP – Paso 7 .....               | 39 |
| <b>Gráfico 33</b> – Conexión al Servidor VPN por PPTP .....                    | 40 |
| <b>Gráfico 34</b> – Conectándose por PPTP .....                                | 41 |
| <b>Gráfico 35</b> – Detalle de Conexión PPTP .....                             | 41 |
| <b>Gráfico 36</b> – Configuración Adicional para L2TP/IPSec.....               | 42 |
| <b>Gráfico 37</b> – Ingreso de Clave Compartida en Conexión L2TP/IPSec.....    | 43 |
| <b>Gráfico 38</b> – Escogiendo L2TP/IPSec en Funciones de Red de Conexión..    | 44 |
| <b>Gráfico 39</b> – Función Config_reg.php .....                               | 46 |
| <b>Gráfico 40</b> – Función DBConnection.php.....                              | 48 |
| <b>Gráfico 41</b> – Página etc.php .....                                       | 49 |

|  |    |
|--|----|
| <b>Gráfico 42</b> – Página index.php.....            | 53 |
| <b>Gráfico 43</b> - json.php.....                    | 72 |
| <b>Gráfico 44</b> – Mante_l2tp.php.....              | 73 |
| <b>Gráfico 45</b> – Mante_pptpd.php.....             | 75 |
| <b>Gráfico 46</b> – pptpd.php .....                  | 76 |
| <b>Gráfico 47</b> –settings.php .....                | 76 |
| <b>Gráfico 48</b> – Users_pptpd.php .....            | 78 |
| <b>Gráfico 49</b> – chap-secretes.php .....          | 79 |
| <b>Gráfico 50</b> – Función Connl2tpd.php .....      | 80 |
| <b>Gráfico 51</b> – Función elimusrPPP.php .....     | 81 |
| <b>Gráfico 52</b> – Función leePptpd.php.....        | 83 |
| <b>Gráfico 53</b> – Función LeeUsrPPP.php .....      | 84 |
| <b>Gráfico 54</b> – Función Nuevousrppp.php .....    | 84 |
| <b>Gráfico 55</b> – Función modificaOptL2tp.php..... | 85 |
| <b>Gráfico 56</b> – Función modificaPptpd.php .....  | 86 |

# MANUAL DE USUARIO

## Instalación del Sistema MyVPN

El Sistema MyVpn para que este operativo, necesitamos de la instalación y configuración de algunos programas, los mismos que detallamos a continuación:

### Sistema Operativo

El sistema operativo para que MyVpn funcione es el Centos Linux Enterprise versión 5.0, el mismo que al momento de instalarlo deben seleccionar el modo de instalación estándar.

### Paquetes Requeridos

Una vez terminada la instalación del Sistema Operativo, debemos continuar con ciertos paquetes o software que hacen falta para que MyVpn funcione correctamente los cuales son:

- ✓ httpd-2.2.3-11.e15\_1.centos.3 (Servidor Apache)
- ✓ php-5.1.6-5.e15 y php-pgsql-5.1.6-5.e15 (PHP V.5)
- ✓ postgresql-8.1.11-1.e15\_1.1 (Base de datos POSTGRES)
- ✓ bind-9.3.3-10.e15 (DNS de Linux)

- ✓ pptpd-1.3.4-1 (Paquete para PPTP)
- ✓ l2tpd-0.69-12jdl (Paquete para L2TP)
- ✓ openswan-2.4.9-2.el5.kb (Paquete para IPsec)
- ✓ sudo-1.6.8p12-10 (Paquete para SUDO)
- ✓ iptables-1.3.5-1.2.1 (Firewall nativo de Linux)

Para mayor información sobre la configuración del Servidor deberá referirse al manual técnico que se encuentra más adelante en este tomo.

### **Explicación de Funcionamiento del Sistema MyVpn**

Con la utilización de este sistema de configuración de servicios de creación de VPN nosotros lo vamos a poder hacer de una manera más fácil y cómoda, a continuación vamos a proceder a describir cada una de las diferentes pantallas en las cuales se basa el presente proyecto.

### **Ingreso al Sistema**

Para ingresar al Sistema MyVPN haremos uso de un browser (preferible Firefox) y colocamos la dirección de la IP de la interface WAN del Servidor VPN añadiendo la ruta de la carpeta en donde están almacenadas las páginas web. En este caso tipeamos:

*<http://192.168.50.2/vpn/index.php>*

mas noticias

## Configuración de VPN

**Inicio**

### Introducción

VPN (Virtual Private Network) red privada virtual. Las redes privadas virtuales crean un túnel unido de un sitio a otro para transferir datos y a ello se le llama en la actualidad los paquetes de datos y viene tripulados de forma que los datos son ilegibles para los extraños. La VPN debe ser capaz de verificar la identidad de los usuarios y restringir el acceso a la VPN a aquellos usuarios que no estén autorizados. Así mismo, debe proporcionar registros estadísticos que muestren quien tuvo acceso, a cual información y cuándo.

### Protocolo PPTP

PPTP (Point-to-Point Tunneling Protocol) es un protocolo desarrollado por Microsoft, U.S. Robotics, Ascend Communications, 3Com/Primary Access, EUI y otras compañías conocidas colectivamente como PPTP Forum, para implementar redes privadas virtuales o VPN.

### Protocolo L2TP/IPSec

L2TP (Layer 2 Tunneling Protocol), de acceso remoto a los clientes a conectarse a una VPN corporativa, solo funciona si se utiliza una conexión segura que es vía IPSec (G. euSza).

#### Inicio de Sesión

Usuario:

Clave:

#### INFORMACION IMPORTANTE

Recordemos que este sistema está abierto para un usuario básico en redes.

COPYRIGHT © 2000 INTCO | VPN@GMAIL.COM | // TML 1.1 | CSS | DISEÑADO POR GUJROO

**Gráfico 1 – Pantalla de Inicio**

Podemos ver que la pantalla de Bienvenida cuenta con una pequeña explicación de los 2 protocolos que se implementan en el sistema.

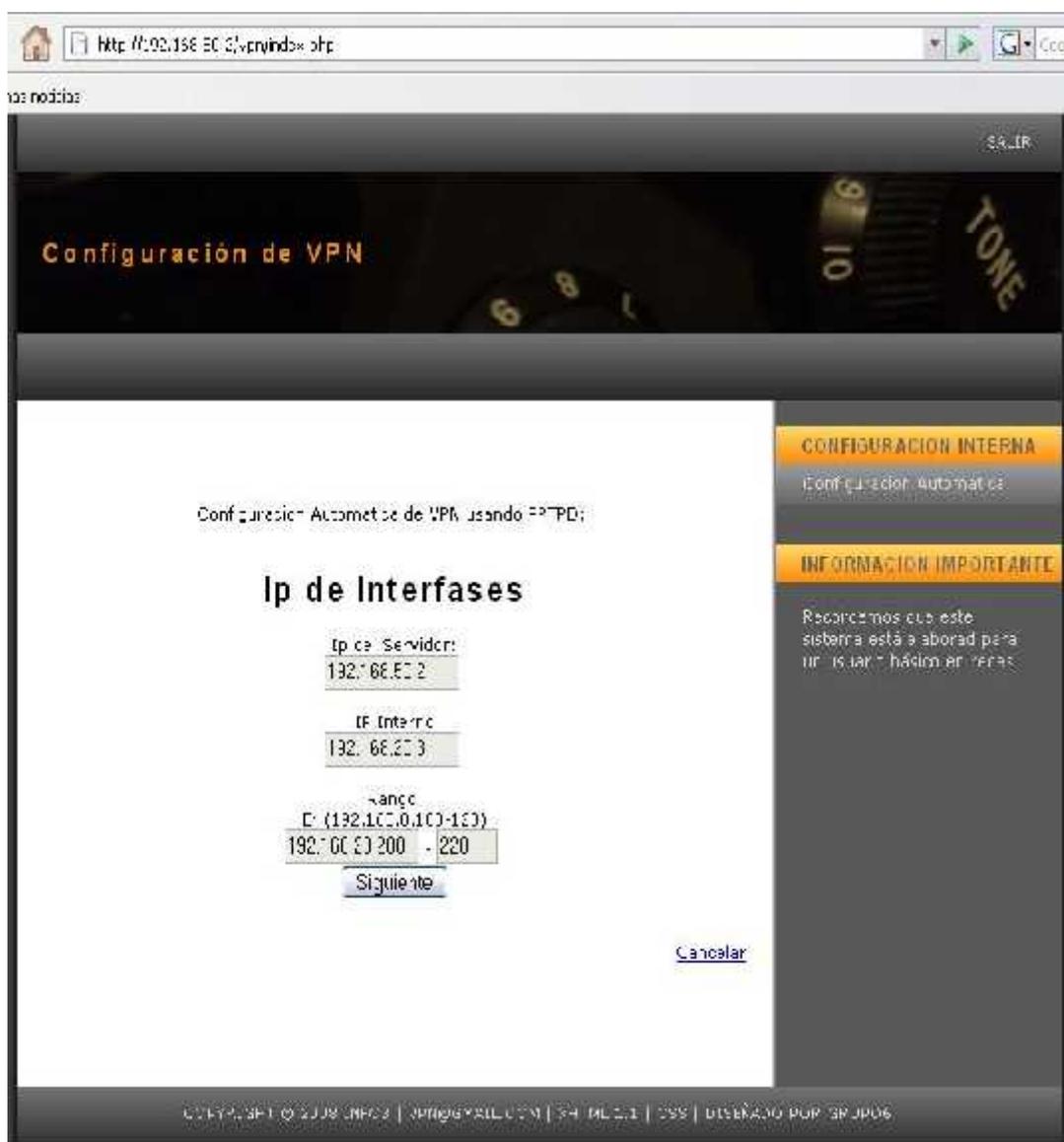


**Gráfico 2 – Pantalla de Bienvenia**

Como podemos darnos cuenta nos aparece una página para el ingreso de los datos del nombre de usuario y clave para su posterior validación y acceso al sistema de configuración de los servicios de creación de VPN. Luego de que se han validado los datos anteriormente ingresados pasamos a la primera pantalla de nuestro sistema donde vamos a encontrar 2 opciones.

## Configuración Automática por el Protocolo PPTP

Nuestro sistema cuenta con una configuración Automática por el protocolo PPTP. En esta opción, el sistema tomará el valor de las IP de las interfaces WAN y LAN (eth0 y eth1) de mi Servidor para poder realizar esta configuración.



**Gráfico 3 – Configuración Automática PPTP Paso 1**

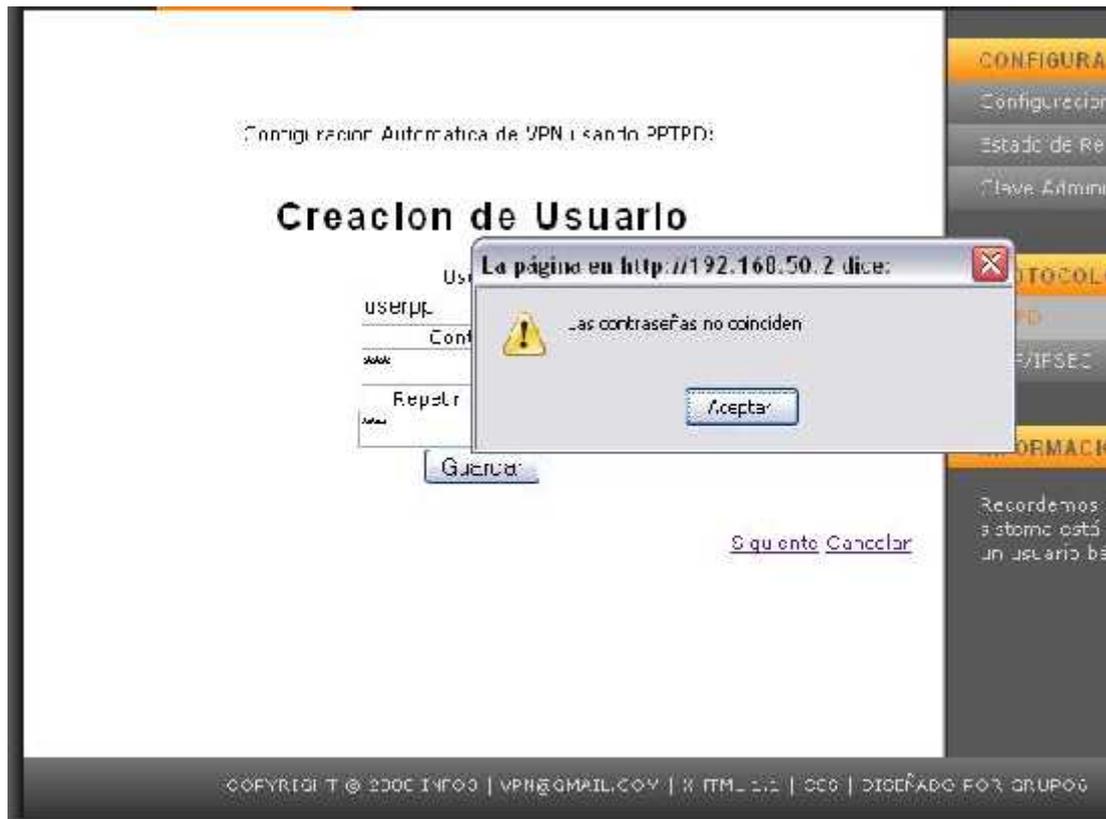
En el gráfico observamos que el sistema valiéndose de la IP de la interfaz de la LAN, asigna un rango por DHCP entre los intervalos de 200 a 220.

Una vez pasada esta opción se pasará a la pantalla “Creación de Usuario”, el Administrador tendrá la opción de Crearlo o No. Si en el momento no deseara crearlo, lo único que hará será dar clic en el Vínculo “Siguiente”



**Gráfico 4a** – Configuración Automática PPTP Paso 2

En caso que el Administrador no coloque contraseñas iguales, el sistema mostrará el siguiente error y no podrá pasar a la siguiente pantalla hasta que las contraseñas sean iguales.



**Gráfico 4b** – Configuración Automática PPTP Paso 2 (Caso de Error)

Una vez que se validaron las contraseñas, el sistema pasa a la última pantalla de Configuración Automática para el levantamiento del protocolo PPTP



**Gráfico 5 – Configuración Automática PPTP Paso 3**

## **Paso a Paso**

### **Por el Protocolo PPTP**

En caso de que el Servidor tenga más de una interfaz de red y su eth1 no corresponde a su IP de LAN sino a su DMZ (por ejemplo), existe la configuración PASO A PASO, que leerá los valores de los archivos de configuración de cada protocolo y permitirá cambiar los valores de las

mismas, tales como IP del Servidor y LAN, y rando por DHCP, como lo muestra el siguiente gráfico:



The screenshot displays the 'Configuración de VPN' interface. At the top, there is a navigation bar with 'INICIO', 'PASO A PASO', and 'CONTENIDO/TEMAS'. The 'PASO A PASO' tab is active. The main content area is titled 'Configuración de VPN' and 'Configuración Automática de VPN usando PPTP'. Below this, the 'Ip de Interfases' section is shown with the following fields:

- Ip del servidor: 192.168.50.3
- IP Interno: 192.168.20.3
- País: (192.168.20C-120)
- 192.168.20.200 220

Buttons for 'Guardar' and 'Cancelar' are visible. On the right side, there is a sidebar with sections: 'CONFIGURACION INTERNA' (containing 'Configuración Automática', 'Estado de Red', and 'Clave Administrador'), 'PROTOCOLOS' (containing 'PPTP' and 'L2TP/IPSEC'), and 'INFORMACION IMPORTANTE' (containing a note: 'Recordemos que este sistema está elaborado para un usuario básico en redes').

**Gráfico 6a** – Paso a Paso por PPTP Paso 1

Así mismo con el hipervínculo Guardar, pasaremos a la pantalla para Creación de Usuario PPTP. También como la pantalla de Configuración Automática, el administrador tiene la opción de crear o no el usuario VPN.



**Gráfico 6b** – Paso a Paso por PPTP Paso 2

Cabe recalcar que el Administrador siempre podrá suspender la acción de cualquier pantalla dando clic en el Hipervínculo “Cancelar”

Después obtendremos una pantalla similar para el levantamiento del Protocolo PPTP.

## Por el Protocolo L2TP/IPSec

El Paso a paso por el protocolo L2TP/IPSec nos guiará también de forma intuitiva a configurar este servicio con la mayor facilidad. Aquí introduciremos las IP de la Interfaz WAN y LAN, una IP que servirá para la interfaz virtual y las IP del rango por DHCP que daremos a los clientes VPN.

The screenshot shows a web-based configuration interface for VPN. The main title is 'Configuración de VPN'. Below the title, there are three tabs: 'INICIO', 'PASO A PASO', and 'RESULTADOS'. The 'PASO A PASO' tab is selected. The main content area is titled 'Configuración automática de VPN usando L2TP/IPSEC' and 'Ip de Interfases'. It contains several input fields and a 'Guardar' button. The fields are: 'Ip de Servidor' (92.68.50.2), 'IP Interfaz de LAN' (92.68.20.3), 'Local IP' (92.68.20.100), and 'Rango' (192.168.20.200 - 192.168.20.250). There is also a 'Cancelar' link at the bottom right. On the right side, there are several menu items: 'CONFIGURACION INTERNA', 'CONFIGURACION ACL/FILOS', 'Estado de Red', 'Otras Administraciones', 'PROTOSOL OS', 'PPTP', 'L2TP/IPSEC', and 'INFORMACION IMPORTANTE'. The footer contains copyright information: 'COPYRIGHT © 2008 INFOS | VPN@EMAIL.COM | 24/7 | 1 | 055 | DESARROLLADO POR EQUIPO'.

**Gráfico 7a – Paso a Paso por L2TP/IPSec Paso 1**

Avanzaremos a la siguiente Pantalla dando clic en el botón “Guardar”. También podremos cancelar la acción dando clic en el hipervínculo “Cancelar”

Resaltamos que el Sistema MyVPN también cuenta con un chequeo de validación de IP, que mostrará un error al Administrador si ingresara una IP inválida.

The screenshot displays the 'Configuración de VPN' interface. The main content area is titled 'Configuración Automática de VPN usando L2TP/IPSEC:' and 'Ip de Interfases'. It contains several input fields for IP addresses, with red error messages indicating they are 'Formato no válido' (invalid format). The fields and their values are:

- IP interfaz -- servidor: 192.168.0.2
- IP interfaz -- LAN: 192.168.0.1
- IP interfaz -- WAN: 192.168.0.6
- Recomendador: 192.168.0.199
- Rango: Ej: 192.168.0.200 - 192.168.0.220, with input 192.168.0.200 - 192.168.0.240

A 'Guardar' button is visible at the bottom of the form. The right sidebar contains navigation options: CONFIGURACION INTERNA, CONFIGURACION AUTOMATICA, Estado de Red, Clase Administrador, PROTOCOLOS, PPTP, L2TP/IPSEC, and INFORMACION IMPORTANTE. The footer includes copyright information: '© 2003 INFOSYS | www.infosys.com | HTML | CSS | DESARROLLO POR GRUPO'.

**Gráfico 7b** – Paso a Paso por L2TP/IPSec (Validación IP)

Superadas las validaciones de IP pasaremos a la pantalla, de creación de Clave Compartida (PSK – Preshared Key). Esta clave es necesaria para la configuración de IPSec, la misma que se utilizarán en los clientes VPN

creados por el protocolo L2TP/IPSec y que no podrán conectarse al Servidor sin la misma.



**Gráfico 8 – Paso a Paso por L2TP/IPSec – Paso 2**

El ingreso de esta clave compartida está sujeta a la validación de contraseñas iguales, la misma que utilizamos para los usuarios VPN. Pasaremos a la siguiente pantalla dando clic en el botón Guardar o dando clic en el hipervínculo “Siguiente”



**Gráfico 9 – Paso a Paso por L2TP/IPSec – Paso 3**

Para la creación de usuarios VPN por el protocolo L2TP/IPSec tendremos el mismo método de funcionamiento que el que utilizamos en los usuarios por PPTP, con la respectiva validación de contraseña.

También si creemos que algo configuramos mal anteriormente, podremos regresar a las pantallas anteriores dando clic en el hipervínculo “Anterior”

Finalmente pasaremos a la pantalla de Levantamiento del protocolo L2TP y del demonio IPsec (OpenSwan). Para el correcto funcionamiento de este protocolo se deben levantar los 2 servicios mencionados.



**Gráfico 10 – Paso a Paso por L2TP/IPSec – Paso 4**

Si alguno de los servicios no se llegara a levantar, es muy probable que el Administrador haya omitido ingresar alguna IP o que los paquetes no estén instalados correctamente, o en su defecto, que se hallan borrado los archivos en el Servidor.



Cuentas de Usuarios: Para crear nuevos usuarios o eliminación de ellos.

Conexiones Activas: Los clientes VPN que estén conctados en ese momento.

Conexiones en el Día: Los clientes VPN que se hayan conectado o desconectado en el transcurso del día.

### Opciones del Servidor

Esta pantalla es la misma que la primera que aparece en la configuración por PPTP, podremos cambiar las IP asignadas a esta configuración si es que el Administrador lo requiere, así también el Rango de DHCP.



**Gráfico 12 – Opciones del Servidor**

## Cuentas de Usuarios

Esta pantalla podremos eliminar los usuarios existentes en el protocolo PPTP o en su defecto también podremos crearlos.

Configuración de VPN

INICIO PASO A PASO **MANTENIMIENTO**

**Cuentas de usuario creadas**

Las cuentas PPTP listadas en esta página se extraen del directorio `/etc/ppp/cnppptp/`, que se utiliza para la autenticación CHAP. Sólo se muestran las de tu servidor local, no las cuentas para hacer conexiones hacia el exterior.

Nuevo Usuario

| Usuarios | Acción   |
|----------|----------|
| user1234 | Eliminar |

[Anterior](#) [Cancelar](#)

COMUNICACION INTERNA  
Configuración Automática  
Estado de Red  
Clave Estándar

PROTOCOLOS  
PPTP

INFORMACION IMPORTANTE  
Recomendamos que este sistema esté configurado para un usuario básico en redes.

Copyright © 2000 INFO - VPN@MAIL.COM | XHTML 1.1 | CSS | Diseñado por GRUPO6

**Gráfico 13 – Cuentas de Usuarios**

Dando clic en el hipervínculo “Nuevo usuario” pasaremos a la pantalla de Creación de Usuario ya explicada anteriormente. Para la eliminación de usuario solo damos clic en el hipervínculo “Eliminar”.

## Conexiones Activas

En esta pantalla, el sistema mostrará los clientes VPN si es que alguno de ellos estuviera en línea. Cuenta con un hipervínculo “Refrescar”, para actualizar la pantalla, en donde aparecerá un nuevo usuario conectado o en su defecto no aparezca ninguno.

The screenshot displays a web interface for VPN configuration. The main content area is titled "Conexiones activas" and contains a table with the following data:

| Fecha | Hora     | Protocolo | Usuario | IP            |
|-------|----------|-----------|---------|---------------|
| 12    | 18:56:43 | ppp2d     | omentes | 192.168.50.31 |
| 12    | 18:57:24 | ppp2d     | usorcp  | 192.168.50.55 |

Below the table are two buttons: "Refrescar" and "Cancelar". The interface also features a navigation menu with "INICIO", "PASO A PASO", and "MANTENIMIENTO" (highlighted). A right sidebar contains sections for "CONFIGURACION INTERNA", "PROTOCOLOS", and "INFORMACION IMPORTANTE".

**Gráfico 14 – Cuentas de Usuarios**

El detalle de la conexión arrojará la Fecha, Hora, Protocolo, Usuario e IP con la que el usuario se conectó en ese momento.

## Conexiones en el Día

Finalmente tendremos la Pantalla “Conexiones en el Día”, aquí podremos visualizar los usuarios que se conectaron en el transcurso del día, así como también en qué momento se desconctaron.

**Historial de conexiones**

**Conectados**

| Fecha | Hora  | Protocolo | Usuario | IP            |
|-------|-------|-----------|---------|---------------|
| 18    | 12:56 | l2pc      |         | 192.168.50.55 |
| 18    | 15:12 | l2pc      |         | 192.168.50.31 |

**Desconectados**

| Fecha | Hora     | Protocolo | Usuario | IP            |
|-------|----------|-----------|---------|---------------|
| 18    | 10:40:00 | l2pc      |         | 192.168.50.01 |
| 18    | 16:37    | l2pc      |         | 192.168.50.55 |

[Verificar Conexión](#)

**Gráfico 15 – Conexiones en el día.**

El Botón “Grabar”, guardará este log de información a nuestra Base de datos para que pueda ser utilizada para auditoría por el administrador si lo deseara.

## Mantenimiento Por L2TP/IPSec



**Gráfico 16 – Mantenimiento por L2TP/IPSec**

Como en el Protocolo L2TP/IPSec, aquí podremos cambiar algún valor específico o usuario si el Administrador lo requiere. También podrá levantar los servicios o bajarlos. Los iconos con que cuenta en esta pantalla son:

Opciones del Servidor: Para cambiar IP en los archivos de este protocolo

Cuentas de Usuarios: Para crear nuevos usuarios o eliminación de ellos.

Clave Compartida: Para cambiar la clave compartida (PSK)

Conexiones Activas: Los clientes VPN que estén conctados en ese momento.

Conexiones en el Día: Los clientes VPN que se hayan conectado o desconectado en el transcurso del día.

## Opciones del Servidor

Esta pantalla es la misma que la primera que aparece en la configuración por L2TP/IPSec, podremos cambiar las IP asignadas a esta configuración si es que el Administrador lo requiere, así también el Rango de DCHP. El chequeo de validación de IP también está presente en esta pantalla.

The screenshot displays the 'Configuración de VPN' interface. At the top, there are navigation tabs: 'INICIO', 'PASO A PASO', and 'MAINTENIMIENTO'. The main content area is titled 'Configuración y Mantenimiento de VPN usando L2TP/IPSEC' and 'Ip de Interfases'. It contains several input fields: 'Ip del Servicio' (142.148.211.2), 'IP de Interfaz de LAN' (192.168.20.1), 'Dirección IP Recomendada' (142.148.211.48), and 'Rango' (92.150.20.200 - 192.168.20.240). There are 'Guardar' and 'Cancelar' buttons. On the right, a sidebar menu includes 'CONFIGURACIÓN INTERNA', 'PROTOCOLOS', 'SERVIDORES', and 'INFORMACIÓN IMPORTANTE'. The bottom of the screen shows a footer with technical details: '© 2009 Cisco Systems, Inc. | http://www.cisco.com | X80-081-141 | CSR | 000-7440-0000-000000000000'.

Gráfico 17 – Opciones del Servidor

## Clave Compartida

En esta pantalla podremos cambiar la Clave Compartida que utiliza IPSec. Cuenta también con la validación de Contraseñas, así como en las anteriores pantallas.



**Gráfico 18 – Clave Compartida**

Guardaremos la clave dando clic en el botón Guardar. Para salir de esta pantalla podremos hacerlo dando clic en el hipervínculo “Cancelar”

## Cuentas de Usuarios

Esta pantalla podremos eliminar los usuarios existentes en el protocolo L2TP/IPSec o en su defecto también podremos crearlos.

**Cuentas de usuario creadas**

Las cuentas PPP L2tp en esta página se extraen del archivo /etc/ppp/chap-secrets, que se utiliza para la autenticación CHAP. Sólo se muestran las de tu servidor l2tpd, no las cuentas para hacer conexiones hacia el exterior.

Nuevo Usuario

| Usuarios                 | Acción                   |
|--------------------------|--------------------------|
| <a href="#">userL2tp</a> | <a href="#">Eliminar</a> |

[Anterior](#) [Cancelar](#)

RECOMENDAMOS que este sistema esté configurado para un usuario básico en redes.

COPYRIGHT © 2000 INFO - VPN@MAIL.COM | XHTML 1.1 | CSS | DISEÑADO POR GRUPO6

**Gráfico 19 – Cuentas de Usuarios**

Dando clic en el hipervínculo “Nuevo usuario” pasaremos a la pantalla de Creación de Usuario ya explicada anteriormente. Para la eliminación de usuario solo damos clic en el hipervínculo “Eliminar”.

## Conexiones Activas

En esta pantalla, el sistema mostrará los clientes VPN si es que alguno de ellos estuviera en línea. Cuenta con un hipervínculo “Refrescar”, para actualizar la pantalla, en donde aparecerá un nuevo usuario conectado o en su defecto no aparezca ninguno.



The screenshot displays a web-based interface for VPN configuration. At the top, there is a navigation menu with options: 'TRÁFICO', 'DESCARGAS', and 'MANTENIMIENTO'. The main content area is titled 'Conexiones activas' and contains a table with the following data:

| Fecha | Hora     | Protocolo | Usuario | IP            |
|-------|----------|-----------|---------|---------------|
| 12    | 18:42:55 | pptp      |         | 192.168.50.55 |
| 12    | 18:45:12 | pptp      |         | 192.168.50.31 |

Below the table, there is a link labeled 'Actualizar Conexión'. The right sidebar contains several sections: 'CONFIGURACION INTERNA' (with sub-items: Configuración Automática, Estado de Red, Conexión de Red), 'PROTOCOLOS' (with sub-items: pptp, l2tp), and 'INFORMACION IMPORTANTE' (with a note: 'Recordemos que este sistema está elaborado para un usuario básico en redes').

**Gráfico 20 – Conexiones Activas**

El detalle de la conexión arrojará la Fecha, Hora, Protocolo e IP con la que el usuario se conectó en ese momento. Debido a las características de este protocolo no es posible obtener el usuario con que se conectó solo la IP.

## Conexiones en el Día

Finalmente (como en el protocolo PPTP) tendremos la Pantalla “Conexiones en el Día”, aquí podremos visualizar los usuarios que se conectaron en el transcurso del día, así como también en qué momento se desconctaron.

**Historial de conexiones**

**Conectados**

| Fecha | Hora     | Protocolo | Usuario | IP            |
|-------|----------|-----------|---------|---------------|
| 22    | 18:12:56 | l2tp      |         | 192.168.50.55 |
| 22    | 18:15:12 | l2tp      |         | 192.168.50.31 |

**Desconectados**

| Fecha | Hora     | Protocolo | Usuario | IP            |
|-------|----------|-----------|---------|---------------|
| 22    | 10:40:00 | l2tp      |         | 192.168.50.01 |
| 22    | 18:16:37 | l2tp      |         | 192.168.50.55 |

[Buscar Conexión](#)

**Gráfico 21** – Conexiones en el día.

El Botón “Grabar”, guardará este log de información a nuestra Base de datos para que pueda ser utilizada para auditoría por el administrador si lo deseara.

## Configuración de la Red

El Sistema MyVPN cuenta con una pantalla en donde me mostrará los valores de IP, Máscara, Puerta de Enlace, asignadas a las interfaces eth0 y eth1. Cabe recalcar que para efectos de una configuración correcta, la interfaz eth0 debe ser de tipo WAN (IP Pública) y la interfaz eth1 será para la LAN de la compañía.



**Gráfico 22 – Configuración de la Red**

## Cambio de Clave de Administrador

La clave del Administrador puede ser cambiada para efectos de Seguridad. Cuenta también con el chequeo de Contraseñas Iguales, para que el usuario pueda recordar con mayor facilidad la nueva contraseña ingresada.



The screenshot displays the 'Configuración de VPN' (VPN Configuration) interface. At the top, there is a navigation bar with 'INICIO', 'PASO A PASO', and 'MANTENIMIENTO'. The main content area is titled 'Cambio de clave' (Change password) and contains two input fields: 'Contraseña' (Password) and 'Repetir contraseña' (Repeat password), both filled with asterisks. Below these fields is a 'Guardar' (Save) button. To the right of the main area is a sidebar menu with sections: 'CONFIGURACION INTERNA' (Internal Configuration) containing 'Configuración Automática', 'Estado de Red', and 'Clave Administrador'; 'PROTOCOLOS' (Protocols) containing 'PPTP' and 'L2TP/ESP'; and 'INFORMACION IMPORTANTE' (Important Information) containing a note: 'Recuerde que este sistema está elaborado para un usuario básico en redes'. A 'Cancelar' (Cancel) link is located below the main area. The footer contains the text: '© 2013 IPSON - VPN@IPSON.COM - HTML 5.1 | CSS | DISEÑO DE ERIPON'.

**Gráfico 23** – Cambio de Clave del Administrador

# MANUAL TÉCNICO

## **Introducción**

El presente manual esta orientado para dar a conocer cuales son los pasos a seguir para poner en funcionamiento nuestro sistema así como también explicar como se encuentra estructurado el sistema.

El Sistema MyVpn se encuentra desarrollado mediante una pagina web la cual trabaja con el lenguaje PHP y algunas validaciones de IP y Contraseñas en JavaScript.

## **Configuración del Servidor**

Realizada la instalación de Centos Linux con los respectivos paquetes adicionales, comenzamos la configuración del servidor VPN con los protocolos PPTP, L2TP/IPSEC. A continuación cómo es la configuración manual de estos archivos, es decir, lo que el sistema hace automáticamente.

## **Archivos de Configuracion de PPTP**

### **Primer archivo a modificar en**

Ubicación: “/etc”

Archivo: “pptpd.conf”

Líneas a configurar:

local ip: 192.168.50.2

remote ip: 192.168.20.100-120

### **Segundo archivo a modificar**

Ubicación: “/etc/ppp”

Archivo: “options.pptpd”

Ahí se le pondrá la IP de mi interfaz 2 (eth1)

ms-dns 192.168.20.3

### **Tercer archivo a modificar**

Para añadir usuarios se añade en el directorio: “/etc/ppp”

Archivo: “chap-secrets”

Se colocará al final de este archivo los usuarios por los siguientes parámetros

| usuario | nombre_protocolo | clave    | ip_asignada |
|---------|------------------|----------|-------------|
| xavier  | pptp             | xavier08 | “*”         |

### **Archivos de Configuración de L2TP/IPSEC**

#### **Primer archivo a modificar**

Ubicación: “/etc/l2tpd”

Archivo: "l2tpd.conf"

Los valores para esas líneas son palabras reservadas serán los siguientes

listen-addr = 192.168.20.3

ip range= 192.168.20.200-192.168.20.210

local ip = 192.168.20.199

Segundo archivo a modificar

Ubicación: "/etc/ppp"

Archivo: "options.l2tpd"

Se cambiará la línea ms-dns

Aquí se colocarán la ip de mi interfaz eth1 que corresponde a mi red LAN

ms-dns 192.168.20.3

**Tercer archivo a modificar**

Ubicación: "/etc"

Archivo: "ipsec.secrets"

Se agregará a la última línea

|              |            |            |                |
|--------------|------------|------------|----------------|
| Ip – eth0    | ip-remotas | tipo-clave | "Contraseña"   |
| 192.168.50.2 | %any:      | PSK        | "clavedemipsk" |

Donde 192.168.50.2 es la ip de mi interfaz eth0

Donde %any quiere decir que cualquier IP se puede conectar.

Donde PSK es el tipo de clave Preshared Key

Donde "clavedemipsk", es la clave de mi PSK (a excepción de las comillas)

#### **Cuarto archivo a modificar:**

Ubicación: "/etc/ppp"

Archivo: "chap-secrets"

Es el mismo archivo del PPTP, pero identificaremos a los usuarios de PPTP con los de L2TP, cambiando el nombre de protocolo como se muestra a continuación:

| usuario | nombre_protocolo | clave   | ip_asignada |
|---------|------------------|---------|-------------|
| david   | l2tp             | david08 | ""          |

#### **Quinto archivo a modificar:**

Nos valdremos de un archivo de ejemplo de configuración para crear nuestro propio archivo:

Ubicación: "/etc/ipsec.d"

Archivo: "L2TP-PSK.conf"

Editaremos las líneas

conn L2TP-PSK

Por: conn seminariog6

left = xxx.xxx.xxx.xxx.

Por: left = 192.168.50.2 (La ip de mi interfaz 0, eth0)

Y cambiaremos la línea

right = xxx.xxx.xxx.xxx.

Por right = %any

El archivo L2TP-PSK.conf lo dejamos intacto y el nuevo archivo con los cambios que hemos realizado lo llamaremos:

“seminariog6.conf” ubicado en: “/etc/ipsec.d/”

El nuevo archivo “seminariog6.conf” quedaría así:

```
conn seminariog6
#
# Configuration for one user with the non-updated Windows 2000/XP.
#
#
# Use a Preshared Key. Disable Perfect Forward Secrecy.
#
authby=secret
pfs=no
#
left=192.168.50.4
#
# Required for original (non-updated) Windows 2000/XP clients.
leftprotoport=17/1701
#
# The remote user.
#
right=%any
rightprotoport=17/1701
#
# Change 'ignore' to 'add' to enable the configuration for this user.
#
auto=add
keyingtries=3
```

**Gráfico 24** – Archivo de Configuración de L2TP/IPSec

### **Reglas de Firewall**

Para que los protocolos puedan implementarse de forma adecuada y el tráfico de información entre los clientes VPN y el Servidor VPN, se deben añadir ciertas políticas de firewall a mi servidor. Esto es posible utilizando el

demonio iptables. A continuación detallamos las líneas de configuración del Firewall.

```
### PARA ABRIR EL PUERTO DE PPP
iptables -A INPUT -i ppp0 -j ACCEPT
iptables -A FORWARD -i ppp0 -o eth0 -j ACCEPT

#### PARA ABRIR EL PUERTO 1723 POR DONDE TRABAJA EL PPTP
iptables -A INPUT -i eth0 -p tcp --dport 1723 -j ACCEPT
iptables -A INPUT -i eth0 -p 47 -j ACCEPT

##### Reglas para el PROTOCLO L2TP /IPSec
iptables -t nat --append PREROUTING -p udp --sport 1701 --dport 1701 -j
DNAT --to-destination 192.168.20.3

#####
# IKE negotiations
iptables -I INPUT -p udp --sport 500 --dport 500 -j ACCEPT
iptables -I OUTPUT -p udp --sport 500 --dport 500 -j ACCEPT

# ESP encrypton and authentication
iptables -I INPUT -p 50 -j ACCEPT
iptables -I OUTPUT -p 50 -j ACCEPT
iptables -I INPUT -p 51 -j ACCEPT
iptables -I OUTPUT -p 51 -j ACCEPT
```

**Gráfico 25** – Reglas de Firewall para protocolos PPTP y L2TP/IPSec

## Procedimiento para la Creación de Conexión de los usuarios que se conectaran a la VPN

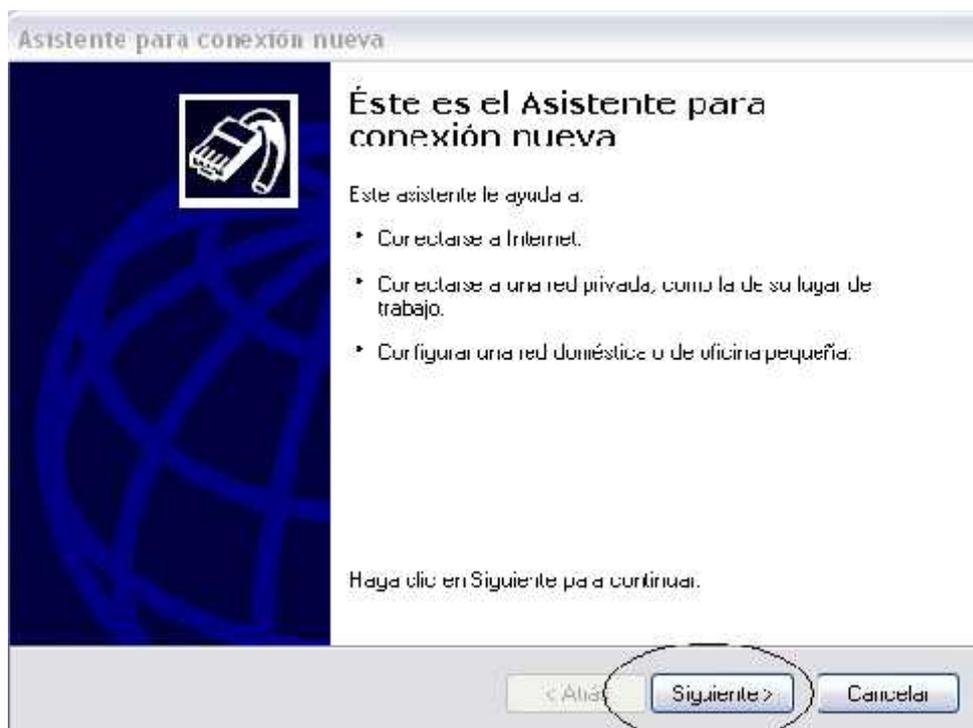
### Creacion de conexión de PPTP

1. Dar click en la opción “Crear una conexión nueva”



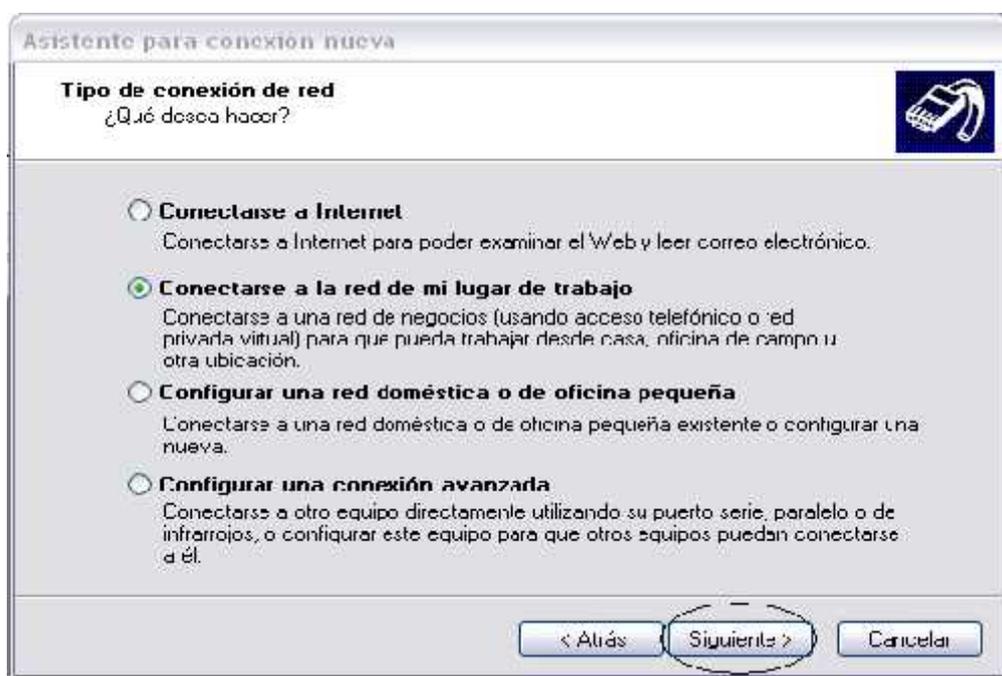
**Gráfico 26** – Conexión de usuarios por PPTP – Paso 1

2. Dar click en siguiente.



**Gráfico 27** – Conexión de usuarios por PPTP – Paso 2

3. Escoger la opción de tipo de conexión de red.



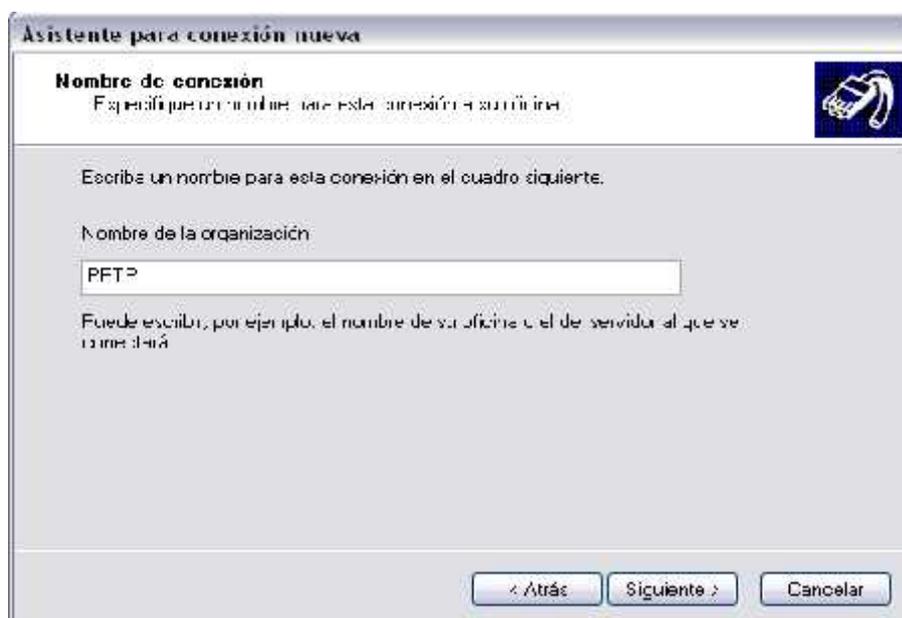
**Gráfico 28** – Conexión de usuarios por PPTP – Paso 3

## 4. Escoger la conexión de red.



**Gráfico 29** – Conexión de usuarios por PPTP – Paso 4

## 5. Ingresamos el nombre de la Conexión, en este caso PPTP



**Gráfico 30** – Conexión de usuarios por PPTP – Paso 5

6. Ingresar la dirección IP del servidor de VPN



Gráfico 31 – Conexión de usuarios por PPTP – Paso 6

7. Finalizar la creación de la conexión.



Gráfico 32 – Conexión de usuarios por PPTP – Paso 7

### Conexión al Servidor VPN por PPTP

Una vez realizada la configuración de Conexión, se procede a conectarse al Servidor VPN, introduciendo el usuario y contraseña, la misma que debe ser facilitada por el Administrador de la VPN



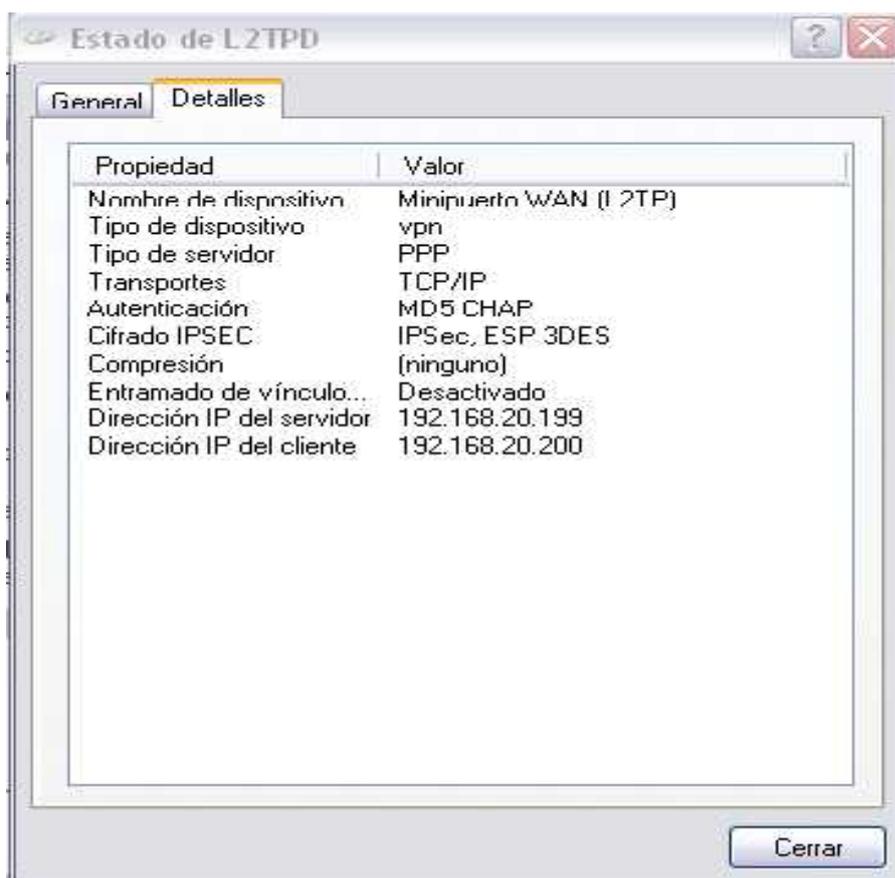
**Gráfico 33** – Conexión al Servidor VPN por PPTP

Al momento de dar clic en “Conectar”, aparecerá la siguiente pantalla que indicará que se está tratando de conectar.



**Gráfico 34 – Conectándose por PPTP**

Siendo la conexión exitosa, entonces nuestro Servidor VPN nos dará una IP de la red de la LAN de la compañía. Con esto ya seremos parte de dicha red remotamente.

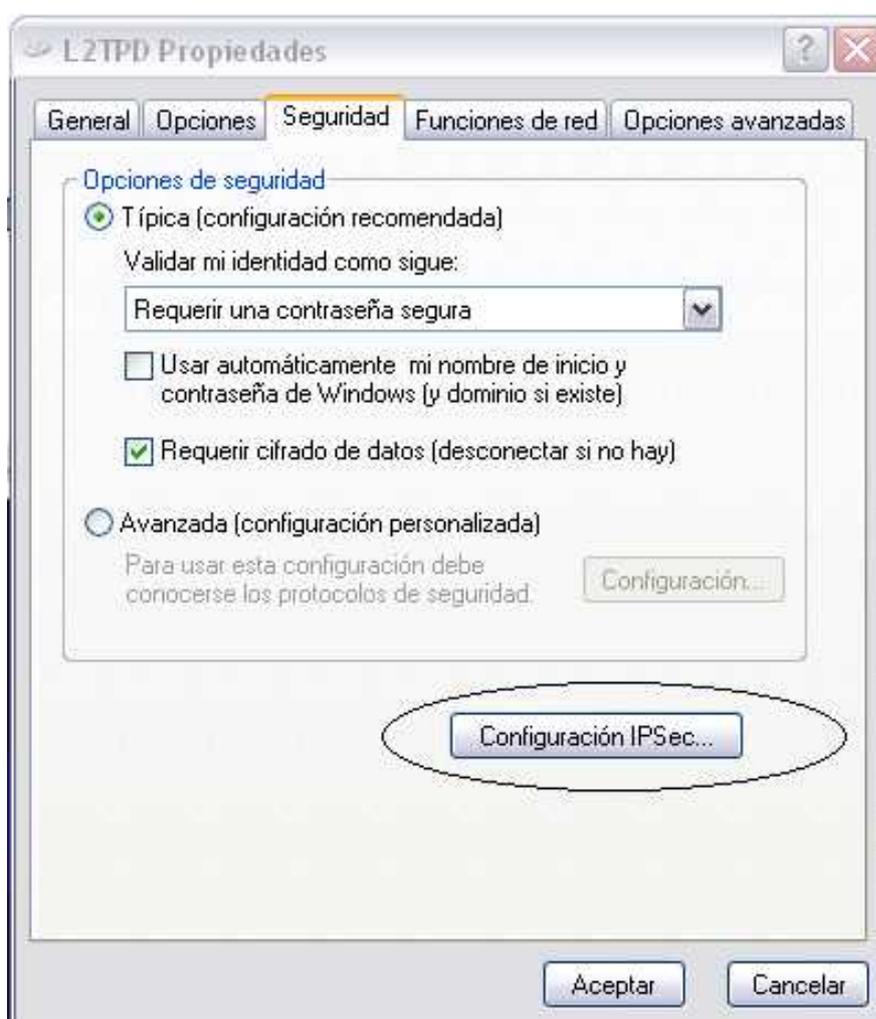


**Gráfico 35 – Detalle de Conexión PPTP**

### Creacion de conexión de L2TP/IPSEC

Para la creación de esta conexión debemos repetir los pasos anteriores desde el punto 1 hasta el punto 7; luego de esto se debe realizar lo siguiente:

- a. En las propiedades de la conexión entramos a la opción de seguridad en el cual damos click en el botón “Configuración IPSEC”.



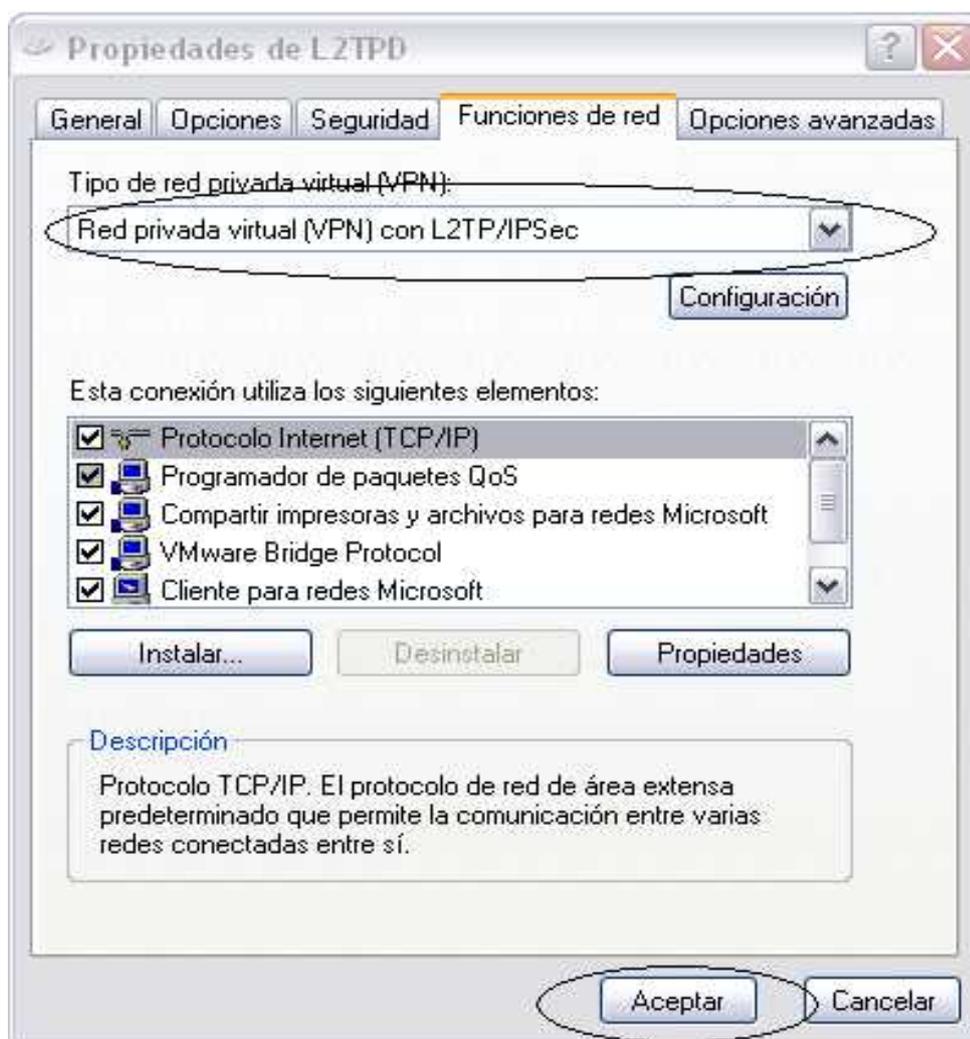
**Gráfico 36** – Configuración Adicional para L2TP/IPSec

- b. En esta pantalla debemos ingresar la Clave Compartida (PSK) la cual se creó en el Servidor VPN mediante el sistema.



**Gráfico 37** – Ingreso de Clave Compartida en Conexión L2TP/IPSec

- c.- En la opción de Funciones de red debemos seleccionar el tipo de red privada virtual, el cual debe ser Red privada virtual (VPN) con L2TP/IPSec.



**Gráfico 38** – Escogiendo L2TP/IPSec en Funciones de Red de Conexión

Ingresamos nuestro usuario y contraseña facilitados por el Administrador de la VPN y nos conectaremos de la misma manera que con un usuario PPTP.

## Código Fuente Principal

Esta función retorna los valores de la configuración básica de la red, la función se llama config\_reg.php. A continuación detallamos el código:

### Función Config\_reg.php

```
<?php
    ob_start();
    session_start();
    require_once("functions/functions.php");
    require_once("JSON.php");
    $json = new Services_JSON();
    $config = $json->decode(vpnFunctions::leerConfigRed());
    if (isset($_SESSION['user'])) {
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<title>Documento sin título</title>
<link rel="stylesheet" type="text/css" href="style/style.css" />
</head>

<body style="background:#FFFFFF">
<br /><br />
<br /><br />
<div align="center">

<br /><br />

<h1>Configuracion de la Red</h1>
<br />
</div>
<div style=" margin-left:100px">
Elementos de la Red<br />
<br />
Interfaz 1 (eth0)<br />
IP:<?php echo $config[0];?><br />
Netmask:<?php echo $config[1];?><br />
Gateway:<?php echo $config[2];?><br />
<br /><br />
Interfaz 2 (eth1)<br />
IP:<?php echo $config[3];?><br />
Netmask:<?php echo $config[4];?><br />
Gateway:<?php echo $config[5];?><br />

</div>
```

```

<div align="right">
<a href="page.php?pagina=cancel" target="_top">Salir</a>
</div>
</body>
</html>
<?php
}
?>

```

**Gráfico 39** – Función Config\_reg.php

### Función DBConnection.php

Esta función nos servirá para realizar la conexión con la base de datos Postgresql, el nombre de la función es: DBConnection.

```

<?php
/*
 * Clase DBConnection para postgresql.
 */
require_once("JSON.php");
class DBConnection {

    private $connection;
    private $result;

    function __construct(){
        $this->connection = pg_pconnect('host=192.168.50.2
dbname=vpn password=vpnadmin user=vpn');
        $status = pg_connection_status($this->connection);
        if ($status !== PGSQL_CONNECTION_OK) {
            throw new Exception("Database connection
failed");
        }
    }

    function query($queryString){
        $this->result = pg_query($this->connection,
$queryString);
    }

    function fetch_array(){
        return pg_fetch_assoc($this->result);
    }
    function fetch_array2(){

```

```

        return pg_fetch_row($this->result);
    }
    function affected_rows(){
        return pg_affected_rows($this->result);
    }

    function rows_returned(){
        return pg_num_rows($this->result);
    }

    function close(){
        pg_close($this->connection);
    }

    function column_type($column_name){
        $field_num = pg_field_num($this->result,
$column_name);
        return pg_field_type($this->result, $field_num);
    }

    function column_types($columns){
        foreach ($columns as $column){
            $types[$column] = $this-
>column_type($column);
        }
        return $types;
    }

    function begin_trans(){
        $this->query("BEGIN TRANSACTION");
    }

    function commit_trans(){
        $this->query("COMMIT");
    }

    function rollback_trans(){
        $this->query("ROLLBACK");
    }

    function clearResults(){
        pg_free_result($this->result);
    }

    function insert($fields, $values,$table,$pk){

    }

    function toJSON(){
        $json = new Services_JSON();
        return $json->encode($this->result);
    }

```

```

function fetch_all(){
    return pg_fetch_all ($this->result);
}
}
?>

```

**Gráfico 40** – Función DBConnection.php

### Función etc.php

```

<?php
require_once("settings.php");
require_once("DBConnection.php");

class etc{
    function check_user($user, $pass,$type){
        $user = "". $user. "";
        if ($type == 'MD5')
            $pass = "".md5($pass). "";
        else
            $pass = "". $pass. "";

        $db_con = new DBConnection();

        $query = "select * from
public.check_user($user,$pass)";
        $db_con->query($query);
        $teams = $db_con->fetch_array2();
        if ($teams[0]=='t')
        {
            $_SESSION['user'] = $user;
            return true;
        }else{
            return false;
        }
    }

    function logout()
    {

```

```

        if (isset($_SESSION['user'])) {
            unset($_SESSION['user']);
            exit;
        }
    }
}
?>

```

**Gráfico 41 – Página etc.php**

## Página Index.php

```

<?php
ob_start();
session_start();
echo '<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">';
echo '<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">';
echo '<head>';
echo '<title>Conexi&oacute;n VPN</title>';
echo '<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />';
echo '<link rel="stylesheet" type="text/css"
href="style/style.css" />';
echo '<link rel="stylesheet" type="text/css"
href="style/orange.css" />';
echo '</head>';
echo '<body>';
echo '<div id="main">';
echo '<div id="links">';
echo '<!--<a href="#">manual pp    </a> | <a href="#">manual
IPSEC</a-->';
if (isset($_SESSION['user']))
    echo '<a href="page.php?pagina=logout">Salir</a>';
echo '</div>';
echo '<div id="logo">';
echo '<h1>Configuraci&oacute;n de VPN</h1>';
echo '</div>';
echo '<div id="menu">';
echo '<ul>';

if (isset($_SESSION['user']))
{
    if (!isset($_SESSION['protocolo']))
    {
        echo '<li><a id="selected"
href="page.php?pagina=cancel">Inicio</a></li>';
        echo '<li><a href="page.php?pagina=wizard">Paso a

```

```

paso</a></li>';
    echo '<li><a
href="page.php?pagina=config">Mantenimiento</a></li>';
    }
    else
    {
        if (isset($_SESSION['accion']))
        {
            switch ($_SESSION['accion'])
            {
                case "wizard":
                    echo '<li><a
href="page.php?pagina=cancel">Inicio</a></li>';
                    echo '<li><a id="selected"
href="page.php?pagina=wizard">Paso a paso</a></li>';
                    echo '<li><a
href="page.php?pagina=config&protocolo=pptpd">Mantenimiento</a></l
i>';
                    break;
                case "config":
                    echo '<li><a
href="page.php?pagina=cancel">Inicio</a></li>';
                    echo '<li><a
href="page.php?pagina=wizard">Paso a paso</a></li>';
                    echo '<li><a id="selected"
href="page.php?pagina=config">Mantenimiento</a></li>';
                    /*
                    case "automatica":
                        echo '<li><a
href="page.php?pagina=cancel">Inicio</a></li>';
                        /*
                        echo '<li><a
href="page.php?pagina=wizard">Paso a paso</a></li>';
                        echo '<li><a
href="page.php?pagina=config">Mantenimiento</a></li>';*/
                        break;
                    //
                    default:
                        //
                        echo '<li><a
href="page.php?pagina=cancel">Inicio</a></li>';
                        /*
                        echo '<li><a
href="page.php?pagina=wizard">Paso a paso</a></li>';
                        echo '<li><a
href="page.php?pagina=config">Mantenimiento</a></li>';*/
                        //
                        break;
                    }
                }
            }
        }
    }
    else
    {
        //echo '<li><a
href="page.php?pagina=cancel">Inicio</a></li>';
    }
}
else
    echo '<li><a href="page.php?pagina=cancel">Inicio</a></li>';
echo '</ul>';

```

```

echo '</div>';
echo '<div id="content">';
echo '<div id="column1">';
echo '<div class="sidebaritem">';
echo '<div class="sbihead">';
echo '    <h1>Estado de red</h1>';
echo '</div>';
echo '<div class="sbilinks">';
echo '<ul>';
//if (isset($_SESSION['user']))
//    echo '<li><a href="page.php?pagina=automatica">Configuracion
Automatica</a></li>';
echo '<li><a href="page.php?pagina=conf_red">Estado de
Red</a></li>';
echo '</ul>';
echo '</div>';
echo '</div>';
if (!(isset($_SESSION['user']))) {
    echo '<div class="sidebaritem">';
    echo '    <div class="sbihead">';
    echo '        <h1>Inicio de session</h1>';
    echo '    </div>';
    echo '    <div class="sbilinks"><form name="login"
action="page.php" method="post">';
    echo '        Usuario:<br />';
    echo '        <input type="text" name="user" id="user"
class="login" /><br />';
    echo '        Clave:<br />';
    echo '        <input type="password" name="pass" id="pass"
class="login"/>';
    echo '        <div align="right">';
    echo '            <input type="hidden" name="pagina" value="login"
/>';
    echo '            <input type="submit" name="submit" id="submit"
value="Entrar" />';
    echo '        </div>';
    echo '    </form>';
    echo '    </div>';
    echo '</div>';
}
if
(isset($_SESSION['user'])&&($_SESSION['accion']!='automatica')&&(i
sset($_SESSION['accion'])))
{
    echo '<div class="sidebaritem">';
    echo '    <div class="sbihead">';
    echo '        <h1>Protocolos</h1>';
    echo '    </div>';
    echo '    <div class="sbilinks">';
    echo '    <ul>';
    if ($_SESSION['protocolo']=='pptpd')
        echo '<li ><a id="actual"
href="page.php?pagina=pptpd">PPTPD</a></li>';
    else

```

```

        echo '<li><a
href="page.php?pagina=pptpd">PPTPD</a></li>';

        if ($_SESSION['protocolo']=='l2tpd')
            echo '<li><a id="actual"
href="page.php?pagina=l2tpd">L2TP/IPSEC</a></li>';
        else
            echo '<li><a
href="page.php?pagina=l2tpd">L2TP/IPSEC</a></li>';
/*
        if ($_SESSION['protocolo']=='ipsec')
            echo '<li><a id="actual"
href="page.php?pagina=ipsec">IPSEC(OPEN VPN)</a></li>';
        else
            echo '<li><a href="page.php?pagina=ipsec">IPSEC(OPEN
VPN)</a></li>';*/
        echo '        </ul>';
        echo '</div>';
        echo '</div>';
    }

echo '        <div class="sidebaritem">';
echo '        <div class="sbihead">';
echo '            <h1>Informacion importante</h1>';
echo '        </div>';
echo '        <div class="sbicontent">';
echo '            <p>';
echo '                Recordemos que para que la configuracion funcione
debemos primero...';
echo '            </p>';
echo '        </div>';
echo '</div>';
echo '<div id="column2" >';
if(isset($_SESSION['pagina']))
    echo '<iframe src="'. $_SESSION['pagina'].'" id="protocolo"
name="protocolo" frameborder="0" width="100%"
height="400px"></iframe>';
else
    if(!isset($_SESSION['protocolo'])&&!isset($_SESSION['accion
']))
        echo '<iframe src="bienvenida.html" id="protocolo"
name="protocolo" frameborder="0" width="100%"
height="730px"></iframe>';
    else
        {
            switch($_SESSION['accion'])
            {
                case 'automatica':
                    echo '<iframe src="wiz_user_pptpd.php"
id="protocolo" name="protocolo" frameborder="0" width="100%"
height="400px"></iframe>';
                    break;
                case 'wizard':

```

```

        switch($_SESSION['protocolo'])
        {
            case 'pptpd':
                echo '<iframe
src="wiz_ips_pptpd.php" id="protocolo" name="protocolo"
frameborder="0" width="100%" height="400px"></iframe>';
                break;
            case 'l2tpd':
                echo '<iframe
src="wiz_ips_l2tpd.php" id="protocolo" name="protocolo"
frameborder="0" width="100%" height="400px"></iframe>';
                break;
        }
        break;
    case 'config':
        switch($_SESSION['protocolo'])
        {
            case 'pptpd':
                echo '<iframe
src="mante_pptpd.php" id="protocolo" name="protocolo"
frameborder="0" width="100%" height="400px"></iframe>';
                break;
            case 'l2tpd':
                echo '<iframe
src="mante_l2tpd.php" id="protocolo" name="protocolo"
frameborder="0" width="100%" height="400px"></iframe>';
                break;
        }
        break;
    }
}
echo '    </div>';
echo ' </div>';
echo ' <div id="footer">';
echo '     copyright &copy; 2008 Info3 | <a
href="#">maxlopezu@gmail.com</a> | <a
href="http://validator.w3.org/check?uri=referer">XHTML 1.1</a> |
<a href="http://jigsaw.w3.org/css-validator/check/referer">CSS</a>
| <a href="http://www.dcarter.co.uk">dise&ntilde;ado por Max</a>';
echo ' </div>';
echo ' </div>';
echo '</body>';
echo '</html>';
?>

```

**Gráfico 42 – Página index.php**

## JSON.php

```
<?php
/* vim: set expandtab tabstop=4 shiftwidth=4 softtabstop=4: */

/**
 * Converts to and from JSON format.
 *
 * JSON (JavaScript Object Notation) is a lightweight data-
interchange
 * format. It is easy for humans to read and write. It is easy for
machines
 * to parse and generate. It is based on a subset of the JavaScript
 * Programming Language, Standard ECMA-262 3rd Edition - December
1999.
 * This feature can also be found in Python. JSON is a text format
that is
 * completely language independent but uses conventions that are
familiar
 * to programmers of the C-family of languages, including C, C++,
C#, Java,
 * JavaScript, Perl, TCL, and many others. These properties make
JSON an
 * ideal data-interchange language.
 *
 * This package provides a simple encoder and decoder for JSON
notation. It
 * is intended for use with client-side Javascript applications
that make
 * use of XMLHttpRequest to perform server communication functions -
data can
 * be encoded into JSON notation for use in a client-side
javascript, or
 * decoded from incoming Javascript requests. JSON format is native
to
 * Javascript, and can be directly eval()'ed with no further
parsing
 * overhead
 *
 * All strings should be in ASCII or UTF-8 format!
 *
 * LICENSE: Redistribution and use in source and binary forms, with
or
 * without modification, are permitted provided that the following
 * conditions are met: Redistributions of source code must retain
the
 * above copyright notice, this list of conditions and the
following
 * disclaimer. Redistributions in binary form must reproduce the
above
 * copyright notice, this list of conditions and the following
disclaimer
 * in the documentation and/or other materials provided with the
```

```

* distribution.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESS OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF
* MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN
* NO EVENT SHALL CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING,
* BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS
* OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR
* TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE
* USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
* DAMAGE.
*
* @category
* @package      Services_JSON
* @author       Michal Migurski <mike-json@teczno.com>
* @author       Matt Knapp <mdknapp[at]gmail[dot]com>
* @author       Brett Stimmerman <brettstimmerman[at]gmail[dot]com>
* @copyright    2005 Michal Migurski
* @license      http://www.opensource.org/licenses/bsd-license.php
* @link         http://pear.php.net/pepr/pepr-proposal-
show.php?id=198
*/

/**
 * Marker constant for Services_JSON::decode(), used to flag stack
state
 */
define('SERVICES_JSON_SLICE', 1);

/**
 * Marker constant for Services_JSON::decode(), used to flag stack
state
 */
define('SERVICES_JSON_IN_STR', 2);

/**
 * Marker constant for Services_JSON::decode(), used to flag stack
state
 */
define('SERVICES_JSON_IN_ARR', 3);

/**
 * Marker constant for Services_JSON::decode(), used to flag stack
state
 */

```

```

define('SERVICES_JSON_IN_OBJ', 4);

/**
 * Marker constant for Services_JSON::decode(), used to flag stack
state
 */
define('SERVICES_JSON_IN_CMT', 5);

/**
 * Behavior switch for Services_JSON::decode()
 */
define('SERVICES_JSON_LOOSE_TYPE', 16);

/**
 * Behavior switch for Services_JSON::decode()
 */
define('SERVICES_JSON_SUPPRESS_ERRORS', 32);

/**
 * Converts to and from JSON format.
 *
 * Brief example of use:
 *
 * <code>
 * // create a new instance of Services_JSON
 * $json = new Services_JSON();
 *
 * // convert a complexe value to JSON notation, and send it to the
browser
 * $value = array('foo', 'bar', array(1, 2, 'baz'), array(3,
array(4)));
 * $output = $json->encode($value);
 *
 * print($output);
 * // prints: ["foo","bar",[1,2,"baz"],[3,[4]]]
 *
 * // accept incoming POST data, assumed to be in JSON notation
 * $input = file_get_contents('php://input', 1000000);
 * $value = $json->decode($input);
 * </code>
 */
class Services_JSON
{
    /**
     * constructs a new JSON instance
     *
     * @param int $use object behavior flags; combine with
boolean-OR
     *
     * possible values:
     * - SERVICES_JSON_LOOSE_TYPE: loose
typing.
     *
     * "{...}" syntax creates
associative arrays

```

```

*                                     instead of objects in
decode().
*                                     - SERVICES_JSON_SUPPRESS_ERRORS:
error suppression.
*                                     Values which can't be
encoded (e.g. resources)
*                                     appear as NULL instead of
*                                     throwing errors.
*                                     By default, a deeply-nested
resource will
*                                     bubble up with an error, so
*                                     all return values
*                                     from encode() should be
checked with isError()
*/
function Services_JSON($use = 0)
{
    $this->use = $use;
}

/**
 * convert a string from one UTF-16 char to one UTF-8 char
 *
 * Normally should be handled by mb_convert_encoding, but
 * provides a slower PHP-only method for installations
 * that lack the multibyte string extension.
 *
 * @param   string  $utf16  UTF-16 character
 * @return  string  UTF-8 character
 * @access  private
 */
function utf162utf8($utf16)
{
    // oh please oh please oh please oh please oh please
    if(function_exists('mb_convert_encoding'))
        return mb_convert_encoding($utf16, 'UTF-8', 'UTF-16');

    $bytes = (ord($utf16{0}) << 8) | ord($utf16{1});

    switch(true) {
        case ((0x7F & $bytes) == $bytes):
            // this case should never be reached, because we
are in ASCII range
            // see:
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
            return chr(0x7F & $bytes);

        case (0x07FF & $bytes) == $bytes:
            // return a 2-byte UTF-8 character
            // see:
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
            return chr(0xC0 | (($bytes >> 6) & 0x1F))
                . chr(0x80 | ($bytes & 0x3F));
    }
}

```

```

        case (0xFFFF & $bytes) == $bytes:
            // return a 3-byte UTF-8 character
            // see:
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
            return chr(0xE0 | (($bytes >> 12) & 0x0F))
                . chr(0x80 | (($bytes >> 6) & 0x3F))
                . chr(0x80 | ($bytes & 0x3F));
        }

        // ignoring UTF-32 for now, sorry
        return '';
    }

/**
 * convert a string from one UTF-8 char to one UTF-16 char
 *
 * Normally should be handled by mb_convert_encoding, but
 * provides a slower PHP-only method for installations
 * that lack the multibyte string extension.
 *
 * @param string $utf8 UTF-8 character
 * @return string UTF-16 character
 * @access private
 */
function utf82utf16($utf8)
{
    // oh please oh please oh please oh please oh please
    if(function_exists('mb_convert_encoding'))
        return mb_convert_encoding($utf8, 'UTF-16', 'UTF-8');

    switch(strlen($utf8)) {
        case 1:
            // this case should never be reached, because we
            // are in ASCII range
            // see:
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
            return $utf8;

        case 2:
            // return a UTF-16 character from a 2-byte UTF-8
            // char
            // see:
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
            return chr(0x07 & (ord($utf8{0}) >> 2))
                . chr((0xC0 & (ord($utf8{0}) << 6))
                    | (0x3F & ord($utf8{1})));

        case 3:
            // return a UTF-16 character from a 3-byte UTF-8
            // char
            // see:
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
            return chr((0xF0 & (ord($utf8{0}) << 4))
                | (0x0F & (ord($utf8{1}) >> 2)))
                . chr((0x80 & (ord($utf8{2}) << 6))
                    | (0x3F & ord($utf8{1})));
    }
}

```

```

        . chr((0xC0 & (ord($utf8{1}) << 6))
            | (0x7F & ord($utf8{2})));
    }

    // ignoring UTF-32 for now, sorry
    return '';
}

/**
 * encodes an arbitrary variable into JSON format
 *
 * @param mixed $var any number, boolean, string, array,
or object to be encoded.
 *
 * see argument 1 to Services_JSON()
above for array-parsing behavior.
 *
 * if var is a string, note that
encode() always expects it
 *
 * to be in ASCII or UTF-8 format!
 *
 * @return mixed JSON string representation of input var or
an error if a problem occurs
 * @access public
 */
function encode($var)
{
    switch (gettype($var)) {
        case 'boolean':
            return $var ? 'true' : 'false';

        case 'NULL':
            return 'null';

        case 'integer':
            return (int) $var;

        case 'double':
        case 'float':
            return (float) $var;

        case 'string':
            // STRINGS ARE EXPECTED TO BE IN ASCII OR UTF-8
FORMAT
            $ascii = '';
            $strlen_var = strlen($var);

            /*
            * Iterate over every character in the string,
            * escaping with a slash or encoding to UTF-8 where
            necessary
            */
            for ($c = 0; $c < $strlen_var; ++$c) {

                $ord_var_c = ord($var{$c});

```

```

switch (true) {
    case $ord_var_c == 0x08:
        $ascii .= '\b';
        break;
    case $ord_var_c == 0x09:
        $ascii .= '\t';
        break;
    case $ord_var_c == 0x0A:
        $ascii .= '\n';
        break;
    case $ord_var_c == 0x0C:
        $ascii .= '\f';
        break;
    case $ord_var_c == 0x0D:
        $ascii .= '\r';
        break;

    case $ord_var_c == 0x22:
    case $ord_var_c == 0x2F:
    case $ord_var_c == 0x5C:
        // double quote, slash, slash
        $ascii .= '\\'.$var{$c};
        break;

    case (($ord_var_c >= 0x20) && ($ord_var_c
<= 0x7F)):
        // characters U-00000000 - U-0000007F
        (same as ASCII)
        $ascii .= $var{$c};
        break;

    case (($ord_var_c & 0xE0) == 0xC0):
        // characters U-00000080 - U-000007FF,
mask 110XXXXX
        // see
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
        $char = pack('C*', $ord_var_c,
ord($var{$c + 1}));
        $c += 1;
        $utf16 = $this->utf82utf16($char);
        $ascii .= sprintf('\u%04s',
bin2hex($utf16));
        break;

    case (($ord_var_c & 0xF0) == 0xE0):
        // characters U-00000800 - U-0000FFFF,
mask 1110XXXX
        // see
http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
        $char = pack('C*', $ord_var_c,
            ord($var{$c + 1}),
            ord($var{$c + 2}));
        $c += 2;
        $utf16 = $this->utf82utf16($char);

```

```

                                $ascii .= sprintf('\u%04s',
bin2hex($utf16));
                                break;

                                case (($ord_var_c & 0xF8) == 0xF0):
                                // characters U-00010000 - U-001FFFFF,
mask 11110XXX
                                // see
                                http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
                                $char = pack('C*', $ord_var_c,
                                                ord($var{$c + 1}),
                                                ord($var{$c + 2}),
                                                ord($var{$c + 3}));
                                $c += 3;
                                $utf16 = $this->utf82utf16($char);
                                $ascii .= sprintf('\u%04s',
bin2hex($utf16));
                                break;

                                case (($ord_var_c & 0xFC) == 0xF8):
                                // characters U-00200000 - U-03FFFFFF,
mask 111110XX
                                // see
                                http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
                                $char = pack('C*', $ord_var_c,
                                                ord($var{$c + 1}),
                                                ord($var{$c + 2}),
                                                ord($var{$c + 3}),
                                                ord($var{$c + 4}));
                                $c += 4;
                                $utf16 = $this->utf82utf16($char);
                                $ascii .= sprintf('\u%04s',
bin2hex($utf16));
                                break;

                                case (($ord_var_c & 0xFE) == 0xFC):
                                // characters U-04000000 - U-7FFFFFFF,
mask 1111110X
                                // see
                                http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
                                $char = pack('C*', $ord_var_c,
                                                ord($var{$c + 1}),
                                                ord($var{$c + 2}),
                                                ord($var{$c + 3}),
                                                ord($var{$c + 4}),
                                                ord($var{$c + 5}));
                                $c += 5;
                                $utf16 = $this->utf82utf16($char);
                                $ascii .= sprintf('\u%04s',
bin2hex($utf16));
                                break;
                                }
}

```

```

        return ''.$ascii.''';

        case 'array':
            /*
            * As per JSON spec if any array key is not an
integer
            * we must treat the the whole array as an object.
We
            * also try to catch a sparsely populated
associative
            * array with numeric keys here because some JS
engines
            * will create an array with empty indexes up to
            * max_index which can cause memory issues and
because
            * the keys, which may be relevant, will be remapped
            * otherwise.
            *
            * As per the ECMA and JSON specification an object
may
            * have any string as a property. Unfortunately due
to
            * a hole in the ECMA specification if the key is a
            * ECMA reserved word or starts with a digit the
            * parameter is only accessible using ECMAScript's
            * bracket notation.
            */

            // treat as a JSON object
            if (is_array($var) && count($var) &&
(array_keys($var) !== range(0, sizeof($var) - 1))) {
                $properties = array_map(array($this,
'name_value'),
                                        array_keys($var),
                                        array_values($var));

                foreach($properties as $property)
                    if(Services_JSON::isError($property))
                        return $property;

                return '{' . join(',', $properties) . '}';
            }

            // treat it like a regular array
            $elements = array_map(array($this, 'encode'),
$var);

            foreach($elements as $element)
                if(Services_JSON::isError($element))
                    return $element;

            return '[' . join(',', $elements) . ']';

        case 'object':

```

```

        $vars = get_object_vars($var);

        $properties = array_map(array($this, 'name_value'),
                                array_keys($vars),
                                array_values($vars));

        foreach($properties as $property)
            if(Services_JSON::isError($property))
                return $property;

        return '{' . join(',', $properties) . '}';

    default:
        return ($this->use & SERVICES_JSON_SUPPRESS_ERRORS)
            ? 'null'
            : new Services_JSON_Error(gettype($var). " can
not be encoded as JSON string");
    }
}

/**
 * array-walking function for use in generating JSON-formatted
name-value pairs
 *
 * @param string $name name of key to use
 * @param mixed $value reference to an array element to be
encoded
 *
 * @return string JSON-formatted name-value pair, like
' "name":value'
 * @access private
 */
function name_value($name, $value)
{
    $encoded_value = $this->encode($value);

    if(Services_JSON::isError($encoded_value))
        return $encoded_value;

    return $this->encode(strval($name)) . ':' . $encoded_value;
}

/**
 * reduce a string by removing leading and trailing comments and
whitespace
 *
 * @param $str string string value to strip of
comments and whitespace
 *
 * @return string string value stripped of comments and
whitespace
 * @access private
 */
function reduce_string($str)

```

```

    {
        $str = preg_replace(array(

            // eliminate single line comments in '// ...' form
            '#^\s*//(.+)\$#m',

            // eliminate multi-line comments in '/* ... */'
form, at start of string
            '#^\s*/\*(.+)\*/#Us',

            // eliminate multi-line comments in '/* ... */'
form, at end of string
            '#/\*(.+)\*/\s*\$#Us'

        ), '', $str);

        // eliminate extraneous space
        return trim($str);
    }

/**
 * decodes a JSON string into appropriate variable
 *
 * @param string $str JSON-formatted string
 *
 * @return mixed number, boolean, string, array, or object
 *             corresponding to given JSON input string.
 *             See argument 1 to Services_JSON() above for
object-output behavior.
 *             Note that decode() always returns strings
 *             in ASCII or UTF-8 format!
 * @access public
 */
function decode($str)
{
    $str = $this->reduce_string($str);

    switch (strtolower($str)) {
        case 'true':
            return true;

        case 'false':
            return false;

        case 'null':
            return null;

        default:
            if (is_numeric($str)) {
                // Lookie-loo, it's a number

                // This would work on its own, but I'm trying
to be
                // good about returning integers where

```

```

appropriate:
    // return (float)$str;

    // Return float or int, as appropriate
    return ((float)$str == (integer)$str)
        ? (integer)$str
        : (float)$str;
} elseif (preg_match('/^(["|\\').*(\\1)$/'s', $str, $m)
&& $m[1] == $m[2]) {
    // STRINGS RETURNED IN UTF-8 FORMAT
    $delim = substr($str, 0, 1);
    $chrs = substr($str, 1, -1);
    $utf8 = '';
    $strlen_chrs = strlen($chrs);

    for ($c = 0; $c < $strlen_chrs; ++$c) {

        $substr_chrs_c_2 = substr($chrs, $c, 2);
        $ord_chrs_c = ord($chrs{$c});

        switch (true) {
            case $substr_chrs_c_2 == '\\b':
                $utf8 .= chr(0x08);
                ++$c;
                break;
            case $substr_chrs_c_2 == '\\t':
                $utf8 .= chr(0x09);
                ++$c;
                break;
            case $substr_chrs_c_2 == '\\n':
                $utf8 .= chr(0x0A);
                ++$c;
                break;
            case $substr_chrs_c_2 == '\\f':
                $utf8 .= chr(0x0C);
                ++$c;
                break;
            case $substr_chrs_c_2 == '\\r':
                $utf8 .= chr(0x0D);
                ++$c;
                break;

            case $substr_chrs_c_2 == '\\\\':
            case $substr_chrs_c_2 == '\\\\\\\\':
            case $substr_chrs_c_2 == '\\\\\\\\\\\\':
            case $substr_chrs_c_2 == '\\\\\/':
                if (($delim == '"' &&
$substr_chrs_c_2 != '\\\\') ||
                    ($delim == "'" &&
$substr_chrs_c_2 != '\\\"')) {
                    $utf8 .= $chrs{++$c};
                }
                break;

```

```

substr($chrs, $c, 6):
character
($c + 2), 2)))
($c + 4), 2));

                                case preg_match('/\\u[0-9A-F]{4}/i',
                                // single, escaped unicode
                                $utf16 = chr(hexdec(substr($chrs,
                                . chr(hexdec(substr($chrs,
                                $utf8 .= $this->utf162utf8($utf16);
                                $c += 5;
                                break;

                                case ($ord_chrs_c >= 0x20) &&
                                ($ord_chrs_c <= 0x7F):
                                $utf8 .= $chrs{$c};
                                break;

                                case ($ord_chrs_c & 0xE0) == 0xC0:
                                // characters U-00000080 - U-
                                000007FF, mask 110XXXXX
                                //see
                                http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
                                $utf8 .= substr($chrs, $c, 2);
                                ++$c;
                                break;

                                case ($ord_chrs_c & 0xF0) == 0xE0:
                                // characters U-00000800 - U-
                                0000FFFF, mask 1110XXXX
                                // see
                                http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
                                $utf8 .= substr($chrs, $c, 3);
                                $c += 2;
                                break;

                                case ($ord_chrs_c & 0xF8) == 0xF0:
                                // characters U-00010000 - U-
                                001FFFFFF, mask 11110XXX
                                // see
                                http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
                                $utf8 .= substr($chrs, $c, 4);
                                $c += 3;
                                break;

                                case ($ord_chrs_c & 0xFC) == 0xF8:
                                // characters U-00200000 - U-
                                03FFFFFF, mask 111110XX
                                // see
                                http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8
                                $utf8 .= substr($chrs, $c, 5);
                                $c += 4;
                                break;

```





```

                                if ($this->use &
SERVICES_JSON_LOOSE_TYPE) {
                                $obj[$key] = $val;
                                } else {
                                $obj->$key = $val;
                                }
                                }

                                } elseif ((($chrs{$c} == '"') || ($chrs{$c}
== "')) && ($top['what'] != SERVICES_JSON_IN_STR)) {
                                // found a quote, and we are not inside
a string
                                array_push($stk, array('what' =>
SERVICES_JSON_IN_STR, 'where' => $c, 'delim' => $chrs{$c}));
                                //print("Found start of string at
{$c}\n");

                                } elseif (($chrs{$c} == $top['delim']) &&
($top['what'] ==
SERVICES_JSON_IN_STR) &&
                                (($chrs{$c - 1} != '\\') ||
($chrs{$c - 1} == '\\') && $chrs{$c
- 2} == '\\')) {
                                // found a quote, we're in a string,
and it's not escaped
                                array_pop($stk);
                                //print("Found end of string at {$c}:
".substr($chrs, $top['where'], (1 + 1 + $c - $top['where']))."\n");

                                } elseif (($chrs{$c} == '[') &&
in_array($top['what'],
array(SERVICES_JSON_SLICE, SERVICES_JSON_IN_ARR,
SERVICES_JSON_IN_OBJ))) {
                                // found a left-bracket, and we are in
an array, object, or slice
                                array_push($stk, array('what' =>
SERVICES_JSON_IN_ARR, 'where' => $c, 'delim' => false));
                                //print("Found start of array at
{$c}\n");

                                } elseif (($chrs{$c} == ']') &&
($top['what'] == SERVICES_JSON_IN_ARR)) {
                                // found a right-bracket, and we're in
an array
                                array_pop($stk);
                                //print("Found end of array at {$c}:
".substr($chrs, $top['where'], (1 + $c - $top['where']))."\n");

                                } elseif (($chrs{$c} == '{') &&
in_array($top['what'],
array(SERVICES_JSON_SLICE, SERVICES_JSON_IN_ARR,

```

```

SERVICES_JSON_IN_OBJ))) {
    // found a left-brace, and we are in an
array, object, or slice
    array_push($stk, array('what' =>
SERVICES_JSON_IN_OBJ, 'where' => $c, 'delim' => false));
    //print("Found start of object at
{$c}\n");

        } elseif (($chrs{$c} == '}') &&
($stop['what'] == SERVICES_JSON_IN_OBJ)) {
    // found a right-brace, and we're in an
object
    array_pop($stk);
    //print("Found end of object at {$c}:
".substr($chrs, $stop['where'], (1 + $c - $stop['where']))."\n");

        } elseif (($substr_chrs_c_2 == '/*') &&
in_array($stop['what'],
array(SERVICES_JSON_SLICE, SERVICES_JSON_IN_ARR,
SERVICES_JSON_IN_OBJ))) {
    // found a comment start, and we are in
an array, object, or slice
    array_push($stk, array('what' =>
SERVICES_JSON_IN_CMT, 'where' => $c, 'delim' => false));
    $c++;
    //print("Found start of comment at
{$c}\n");

        } elseif (($substr_chrs_c_2 == '*/') &&
($stop['what'] == SERVICES_JSON_IN_CMT)) {
    // found a comment end, and we're in
one now
    array_pop($stk);
    $c++;

    for ($i = $stop['where']; $i <= $c;
++$i)
        $chrs = substr_replace($chrs, ' ',
$i, 1);

    //print("Found end of comment at {$c}:
".substr($chrs, $stop['where'], (1 + $c - $stop['where']))."\n");

        }
    }

    if (reset($stk) == SERVICES_JSON_IN_ARR) {
        return $arr;

    } elseif (reset($stk) == SERVICES_JSON_IN_OBJ)
{
        return $obj;
}

```

```

    }
}

}

}

/**
 * @todo Ultimately, this should just call PEAR::isError()
 */
function isError($data, $code = null)
{
    if (class_exists('pear')) {
        return PEAR::isError($data, $code);
    } elseif (is_object($data) && (get_class($data) ==
'services_json_error' ||
is_subclass_of($data,
'services_json_error')) {
        return true;
    }

    return false;
}

if (class_exists('pear_error')) {

    class Services_JSON_Error extends PEAR_Error
    {

        function Services_JSON_Error($message = 'unknown error',
$code = null,
$mode = null, $options = null,
$userinfo = null)
        {
            parent::PEAR_Error($message, $code, $mode, $options,
$userinfo);
        }
    }

} else {

    /**
     * @todo Ultimately, this class shall be descended from
PEAR_Error
     */

    class Services_JSON_Error
    {
        function Services_JSON_Error($message = 'unknown error',
$code = null,
$mode = null, $options = null,

```

```

$userinfo = null)
    {
        }
    }
}
?>

```

**Gráfico 43 - json.php**

### Mante\_l2tp.php

```

<?php
    ob_start();
    session_start();
    if (isset($_SESSION['user'])) {
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<title>Documento sin título</title>
<link rel="stylesheet" type="text/css" href="style/style.css" />
</head>

<body style="background:#FFFFFF">
<br /><br />
<br /><br />
<div align="center">
Configuracion de VPN usando L2TP/IPSEC:
<br /><br />
<br /><br />
<table>
<tr>
<td><center></center></td><td>&nbsp;</td>
<td><center></center></td><td>&nbsp;</td>
<td><center></center></td>
</tr>
<tr>
<td><a href="page.php?pagina=wiz_ips_l2tpd" target="_top">Opciones
del servidor</a></td><td>&nbsp;</td>
<td><a href="page.php?pagina=wiz_claves_l2tpd" target="_top">Claves
Compartidas</a></td><td>&nbsp;</td>
<td><a href="page.php?pagina=wiz_user_l2tpd" target="_top">Cuentas

```

```

de Usuarios</a></td>
</tr>
</table>
<br /><br />
<form name="levantarL2tp" id="levantarL2tp" action="page.php"
method="post" target="_top">
<?
if (!isset($_SESSION['servicio']))
{
    echo '<input type="hidden" name="pagina" value="levantarL2tp"
/>';
    echo '<input type="submit" value="Levantar L2TP" />';
}
else
{
    if ($_SESSION['servicio']=='l2tp')
    {
        echo '<input type="hidden" name="pagina"
value="detenerL2tp" />';
        echo '<input type="submit" value="Detener L2TP" />';
    }
}
?>
</form>
<form name="levantarIpsec" id="levantarIpsec" action="page.php"
method="post" target="_top">
<?php
if (!isset($_SESSION['servicio2']))
{
    echo '<input type="hidden" name="pagina" value="levantarIpsec"
/>';
    echo '<input type="submit" value="Levantar IPSEC" />';
}
else
{
    if ($_SESSION['servicio2']=='ipsec')
    {
        echo '<input type="hidden" name="pagina"
value="detenerIpsec" />';
        echo '<input type="submit" value="Detener IPSEC" />';
    }
}
?>
</form>
</div>
<br />
<br />
<div align="right">
<!--<a href="index.php" target="_top">Cancelar</a>-->
</div>
</body>
</html>

```

**Gráfico 44 – Mante\_l2tp.php**

**Mante\_pptpd.php**

```

<?php
    ob_start();
    session_start();
    if (isset($_SESSION['user'])) {
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<title>Documento sin título</title>
<link rel="stylesheet" type="text/css" href="style/style.css" />
</head>

<body style="background:#FFFFFF">
<br /><br />
<br /><br />
<div align="center">
Configuracion de VPN usando PPTPD:
<br /><br />
<br /><br />
<table>
<tr>
<tr>
<td><center></center></td><td>&nbsp;</td>
<td><center></center></td>
<td><center></center></td>
</tr>
<tr>
<td><a href="page.php?pagina=wiz_pptpd" target="_top">Opciones del
servidor</a></td><td>&nbsp;</td>
<td><a href="page.php?pagina=user_pptpd" target="_top">Cuentas de
Usuarios</a></td>
<td><a href="page.php?pagina=conexionesPptpd"
target="_top">Conexiones activas</a></td>
</tr>
</table>
<br /><br />
<form name="levantarPppt" id="levantarPppt" action="page.php"
method="post" target="_top">
<?
if (!isset($_SESSION['servicio']))
{
    echo '<input type="hidden" name="pagina" value="levantarPppt"
/>';
    echo '<input type="submit" value="Aplicar Configuracion" />';
}
else

```

```

{
    echo '<input type="hidden" name="pagina"
value="detenerPppt" />';
    echo '<input type="submit" value="Detener PPTP" />';
}
?>

</form>
</div>
<br />
<br />
<div align="right">
<!--<a href="index.php" target="_top">Cancelar</a-->
</div>
</body>
</html>
<?php
}
?>

```

**Gráfico 45 – Mante\_pptpd.php**

## pptpd.php

```

<?php
    ob_start();
    session_start();
    if (isset($_SESSION['user'])) {
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Documento sin título</title>
<link rel="stylesheet" type="text/css" href="style/style.css"
/>
</head>

<body style="background:#FFFFFF">
<br /><br />
<br /><br />
<div align="center">
Configuración de VPN usando PPTPD:
<br /><br />

```

```

<br /><br />
<table>
<tr>
<td><center></center></td><td>&nbsp;</td>
<td><center></center></td>
</tr>
<tr>
<td><a href="page.php?pagina=wiz_pptpd"
target="_top">Configuraci&oacute;n
asistida</a></td><td>&nbsp;</td>
<td><a href="page.php?pagina=config&protocolo=pptpd"
target="_top">Configuracion Manual</a></td>
</tr>
</table>
</div>
<br />
<br />
<div align="right">
<!--<a href="index.php" target="_top">Cancelar</a-->
</div>
</body>
</html>
<?php
}
?>

```

**Gráfico 46 – pptpd.php**

## Settings.php

```

<?php
$postgres_user = "vpn";
$postgres_pass = "vpnadmin";
$postgres_server = "localhost";
$postgres_dbname = "vpn";

$dsn='host=.'.$postgres_server.' dbname=.'.$postgres_dbname.'
user=.'.$postgres_user.' password=.'.$postgres_pass;
?>

```

**Gráfico 47 – settings.php**

## Users\_pptpd.php

```

<?php
    ob_start();
    session_start();
    require_once("functions/functions.php");
    require_once("JSON.php");
    if (isset($_SESSION['user'])) {
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Documento sin título</title>
<link rel="stylesheet" type="text/css" href="style/style.css"
/>
/>
</head>

<body style="background:#FFFFFF">
<br /><br />
<br /><br />
<div align="center">

<br /><br />

<h1>Cuentas de usuario creadas</h1>
Las cuentas PPP listadas en esta página se extraen del fichero
/etc/ppp/chap-secrets, <br />
que se utiliza para la autenticación CHAP. Sólo se muestran
las de tu servidor pptpd, <br />
no las cuentas para hacer conexiones hacia el exterior.</div>
<br /><center>
<a href="wiz_user_pptpd.php" target="_self">Nuevo
Usuario</a><br/><br/>
<table border="0" style="border:solid #666666 1px"
cellpadding="0" cellspacing="0" width="400px">
<tr style="background-color:#003399; color:#FFFFFF; font-
size:12px"><td><strong>Usuarios</strong></td><td><strong>Accion
</strong></td></tr>
<?php
    $json = new Services_JSON();
    $usuarios = $json-
>decode(vpnFunctions::usuarioLeePpp("/etc/ppp/chap-secrets",
$_SESSION['protocolo']));
    if ($usuarios!=null)
        foreach ($usuarios as $n_usuario => $usuario)
            echo '<tr><td><a
href="wiz_user_'.$_SESSION['protocolo'].'.php?name='.$usuario.'"
">'.$usuario.'</a></td><td><a
href="page.php?pagina=eliminaUsuario&user='.$usuario.'"
target="_top">Eliminar</a></td></tr>';

```

```

?>
</table>
</center>
<br />
<div align="right">
<?php
if ($_SESSION['protocolo']=='pptpd')
    echo '<a href="wiz_user_pptpd.php"
target="_self">Anterior</a>';
else
    echo '<a href="wiz_user_l2tpd.php"
target="_self">Anterior</a>';
?>&nbsp;
<?php
if ($_SESSION['accion']=='config')
    echo '<a href="page.php?pagina=config"
target="_top">Cancelar</a>';
else
    echo '<a href="page.php?pagina=cancel"
target="_top">Cancelar</a>';
?>

</div>
</body>
</html>
<?php
}
?>

```

**Gráfico 48 – Users\_pptpd.php**

### **Función Chap-secrets.php**

```

<?php
function existeUsuarioPPP($Archivo, $usuario){
    // Backup file
    $var=system('sudo /bin/chmod o+r '.$Archivo,$retal);
    $f_arr = file($Archivo);

    // Open that same file in write mode (blanking it)
    //$fh = fopen($Archivo,'w');

    # Two methods could be used here - a for loop or a foreach
    loop. Below is a for loop

    // Lets count the number of lines
    $count = count($f_arr);

    // Start the for loop

```

```

for($i=0; $i<$count; $i++)
{
    //echo $f_arr[$i].'<br>';
    if ($usuario==substr($f_arr[$i],0,strlen($usuario) ) )
    // if ($usuario==substr($f_arr[$i],0,6) )
    {
        return true;
    }
    //$f_arr[$i] = 'Todo bien!'.chr(10);

    $var=exec('sudo /bin/chmod o-r '.$Archivo,$retal);

    // Now write out the line into the blank file
    //fwrite($fh,$f_arr[$i]); # Note that there is no newline
character since i didnt trim() the line.
}

}

if (existeUsuarioPPP("/etc/ppp/chap-secrets2","max"))
    echo 'Lo encontro';
else
    echo 'No lo encontro';
?>

```

**Gráfico 49 – chap-secretes.php**

## Función Connl2tpd.php

```

<?php
function modifacaconnL2tp($Archivo,$nombreconn,$IP){
    // Backup file

    $var=system('sudo /bin/chmod o+wx /etc/ipsec.d',$retal);
    $var=system('sudo /bin/chmod o+wr '.$Archivo,$retal);
    //$f_arr = file($Archivo);

    // Open that same file in write mode (blinking it)
    $fh = fopen($Archivo,'w');

```

```

# Two methods could be used here - a for loop or a foreach loop.
Below is a for loop

// Lets count the number of lines
// $count = count($f_arr);
// echo $count;
// Start the for loop
// echo $f_arr[10];
for($i=0; $i<9; $i++)
{ //      $f_arr[$i]='Xavier'.chr(10);

    if ($i==0) $linea='conn ' . $nombreconn.chr(10);

    if ($i==1) $linea='  authby=secret'.chr(10);
    if ($i==2) $linea='  pfs=no'.chr(10);

    if ($i==3) $linea='  left=' . $IP.chr(10);

    if ($i==4) $linea='  leftprotoport=17/1701'.chr(10);

    if ($i==5) $linea='  right=%any'.chr(10);
    if ($i==6) $linea='  rightprotoport=17/1701'.chr(10);

    if ($i==7) $linea='  auto=add'.chr(10);
    if ($i==8) $linea='  keyingtries=3'.chr(10);

    // $f_arr[$i] = 'Todo bien!'.chr(10);

    // Now write out the line into the blank file
    fwrite($fh,$linea); # Note that there is no newline character
    since i didnt trim() the line.
}
$var=system('sudo /bin/chmod o-wr ' . $Archivo,$retal);
$var=system('sudo /bin/chmod o-wrx /etc/ipsec.d');
}

modificaconnL2tp("/etc/ipsec.d/no_oe.conf", "grupo6", "192.168.50.4")
?>

```

**Gráfico 50 – Función Connl2tpd.php**

## Función elimUsrPPP.php

```
<?php
function elimUsrPPP($Archivo, $usuario){
    // Backup file
    $var=system('sudo /bin/chmod o+rw '.$Archivo,$retal);
    $f_arr = file($Archivo);

    // Open that same file in write mode (blanking it)
    $fh = fopen($Archivo,'w');

    # Two methods could be used here - a for loop or a foreach
    loop. Below is a for loop

    // Lets count the number of lines
    $count = count($f_arr);
    $ban=0;
    // Start the for loop
    for($i=0; $i<$count; $i++)
    {
        if ($usuario==substr($f_arr[$i],0,strlen($usuario) ) )
        {
            //$i++;
            $f_arr[$i]="";

            //$f_arr[$i]=$usuario.' * '.$password.' "*"
            '.chr(10);

        }

        fwrite($fh,$f_arr[$i]); # Note that there is no newline
        character since i didnt trim() the line.
    }
    $var=system('sudo /bin/chmod o-rw '.$Archivo,$retal);
}
?>
```

**Gráfico 51** – Función elimusrPPP.php

## Función LeePtpd.php

```

<?php
function leePtpd($Archivo){
    // Backup file
    // $var=system('sudo /bin/chmod o+r '.$Archivo,$retal);
    $f_arr = file($Archivo);

    // Open that same file in write mode (blanking it)

    # Two methods could be used here - a for loop or a foreach
    loop. Below is a for loop

    // Lets count the number of lines
    $count = count($f_arr);
    $ban=0;
    $ban1=0;
    // Start the for loop
    for($i=0; $i<$count; $i++)
    {
        //echo $f_arr[$i].<br>';

        if ("#localip"==substr($f_arr[$i],0,8) or
"#localip"==substr($f_arr[$i],0,7))
            if ($ban==0)
                {
                    if ("#"==substr($f_arr[$i],0,1))
                        echo
substr($f_arr[$i],9,50).<br>';
                    else
                        echo
substr($f_arr[$i],8,50).<br>';

                        $ban++;
                }
            // $f_arr[$i] = 'Todo bien!'.chr(10);

            if ("#remoteip"==substr($f_arr[$i],0,9) or
"#remoteip"==substr($f_arr[$i],0,8))
                if ($ban1==0 )
                    {
                        if ("#"==substr($f_arr[$i],0,1))
                            echo
substr($f_arr[$i],10,50).<br>';
                        else
                            echo
substr($f_arr[$i],9,50).<br>';

                            $ban1++;
                    }
    }
}

```

```

        // $f_arr[$i] = 'Todo bien!'.chr(10);

        // Now write out the line into the blank file
    }
}
?>

```

**Gráfico 52 – Función leePtpd.php**

### Función LeeUsrPPP.php

```

<?php
function leeUsersPpp($Archivo, $protocolo){
    // Backup file
    $var=system('sudo /bin/chmod o+r '.$Archivo,$retal);
    $f_arr = file($Archivo);

    // Open that same file in write mode (blinking it)
    //$fh = fopen($Archivo,'w');

    # Two methods could be used here - a for loop or a foreach
    loop. Below is a for loop

    // Lets count the number of lines
    $count = count($f_arr);

    // Start the for loop
    for($i=0; $i<$count; $i++)
    {

        $posicion=stripos($f_arr[$i],$protocolo);

        if ($posicion==true)

        {
            $a=strlen($f_arr[$i])-1;

            $user=substr($f_arr[$i],0,$posicion-1);
            //$pos2=strrpos($f_arr[$i]," ");

            $result= $a - ($posicion + 9);
            $passwd=substr($f_arr[$i],$posicion + 6,$result);

            echo 'El usuario '.$user.' Passwd:

```

```

' . $passwd. ' <br> ' ;
}
//fwrite($fh,$f_arr[$i]); # Note that there is no newline
character since i didnt trim() the line.
}
    $var=exec('sudo /bin/chmod o-r ' . $Archivo,$retal);
?>

```

**Gráfico 53 – Función LeeUsrPPP.php**

### Función Nuevousrppp.php

```

<?php
function nuevoUsrPPP($Archivo, $usuario, $protocolo
,$password){
    // Backup file
    $var=system('sudo /bin/chmod o+rw ' . $Archivo,$retal);
    $fp = fopen($Archivo, "a+");

    if ($fp){
    //while ($linea= fgets($fp,1024)){
    //    echo $linea;
    //}
    // $cadenauser=$_GET["Us"].' * ' . $_GET["Uc"].' "*"
    '.chr(10);
    $cadenauser=$usuario.' ' . $protocolo.' ' . $password.'
    "*"'.chr(10);
    fwrite($fp,$cadenauser);
    }
    $var=system('sudo /bin/chmod o-rw ' . $Archivo,$retal);

}

nuevoUsrPPP("/etc/ppp/chap-secrets", "mayra", "pptpd", "mayra08");
?>

```

**Gráfico 54 – Función Nuevousrppp.php**

## Función modificaOptL2tp.php

```

<?php
function modificaOptL2tp($Archivo, $IP1){
    // Backup file
    $var=system('sudo /bin/chmod o+w '.$Archivo,$retal);
    $f_arr = file($Archivo);

    // Open that same file in write mode (blanking it)
    $fh = fopen($Archivo,'w');

    # Two methods could be used here - a for loop or a foreach
    loop. Below is a for loop

    // Lets count the number of lines
    $count = count($f_arr);
    $ban=0;
    // Start the for loop
    for($i=0; $i<$count; $i++)
    {

        if ("ms-dns"==substr($f_arr[$i],0,6))
        {
            if ($ban==0)
            {
                $f_arr[$i]='ms-dns '.$IP1.chr(10);
                echo 'Se encontro en la linea '.$i.'
'.'. $f_arr[$i];
                $ban++;
            }
            else
            $f_arr[$i]='';
        }

        if ("ms-wins"==substr($f_arr[$i],0,7)) $f_arr[$i]='';

        // if ("ms-wins"==substr($f_arr[$i],0,7))
        $f_arr[$i]='';
    }
    $var=system('sudo /bin/chmod o-w '.$Archivo,$retal);
}
?>

```

**Gráfico 55** – Función modificaOptL2tp.php

## Función modificaOptPptpd.php

```

<?php
function modificaOptPptpd($Archivo, $IP1){
    // Backup file
    $var=system('sudo /bin/chmod o+w '.$Archivo,$retal);
    $f_arr = file($Archivo);

    // Open that same file in write mode (blanking it)
    $fh = fopen($Archivo,'w');

    # Two methods could be used here - a for loop or a foreach
    loop. Below is a for loop

    // Lets count the number of lines
    $count = count($f_arr);
    $ban=0;
    // Start the for loop
    for($i=0; $i<$count; $i++)
    {

        if ("#ms-dns"==substr($f_arr[$i],0,7) or "ms-
        dns"==substr($f_arr[$i],0,6))
            if ($ban==0)
                {
                    $f_arr[$i]='ms-dns '.$IP1.chr(10);
                    echo 'Se econtro en la linea '.$i.'
                '. $f_arr[$i];
                    $ban++;
                }
            fwrite($fh,$f_arr[$i]); # Note that there is no newline
            character since i didnt trim() the line.
        }
    }
    $var=system('sudo /bin/chmod o-w '.$Archivo,$retal);
}
?>

```

**Gráfico 56 – Función modificaPptpd.php**