

**SISTEMA DE GESTION DE LOS AMARRES ENTRE LOS
TRANSFORMADORES DE DISTRIBUCIÓN Y LOS USUARIOS DE LA
EMPRESA CHC - SUN RPC**

MANUAL DE INSTALACION



Integrantes

CRISITAN CAMILO ERAZO

JULIO CÉSAR LUNA

Profesor

ING. PABLO MAGÉ IMBACHÍ

Universidad del cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Programa de ingeniera de sistemas

Curso: Laboratorio de sistemas distribuidos

Popayán marzo 2015

TABLA DE CONTENIDO

1. Introducción
2. Requerimientos del Sistema
3. Descripción de la documentación entregada
4. Descripción de la estructura de directorios, descripción de los archivos de soporte
5. Descripción de los archivos fuente (Interfaces, archivos cliente, archivos servidor)
6. Descripción del proceso de compilación
7. Descripción del proceso de Ejecución

1. INTRODUCCIÓN

Este manual permite al administrador del sistema conocer los requerimientos y pasos a seguir necesarios para el correcto funcionamiento e instalación del proyecto desarrollado el cual permite la gestión de información obtenida por sensores ubicados en residencias que capturan la información de lectura del consumo de energía.

El software fue desarrollado de manera que un administrador (cliente) y dos servidores (local y esclavo) interactuaran entre si haciendo que el cliente envíe peticiones al servidor local y este a su vez le envía la petición al servidor esclavo quien ejecute las funciones necesarias para el funcionamiento del programa, permitiendo las diferentes operaciones que necesitan desarrollar los diferentes actores del programa.

2. REQUERIMIENTOS DEL SISTEMA:

Para realizar el proceso de esta aplicación, el sistema distribuido se podrá ejecutar en computadores que tengan siguientes características:

Sistema Operativo:

Sistema Operativo Linux Debian 7.8

Tecnología utilizada:

Sun RPC

Lenguaje de programación utilizado:

Lenguaje C

Repositorios Necesarios:

- Portmapper
- Libc6-dev
- gcc

Para verificar la existencia en el computador de lo anterior:

- Portmapper: en la consola de Debian ingresamos
rpcinfo -p localhost
- Libc6-dev: entrar en el synaptic de Debian como súper usuario, hacer una búsqueda del paquete y verificar si está marcado como instalado.
- gcc: entrar en el synaptic de Debian como súper usuario, hacer una búsqueda del paquete y verificar si está marcado como instalado.

Lo cual debería de arrojarlos como información que está escuchando por el puerto 111 del tcp y del udp.

Hardware:

Las siguientes características de hardware fueron sobre las que se desarrolló y probó el programa.

- Pentium(R) Dual-Core
- Procesador 210 GHz
- RAM 3,00 GB

3. DESCRIPCION DE LA DOCUMENTACION ENTREGADA

La documentación del sistema distribuido está compuesta por tres manuales: Manual de Instalación, Manual Técnico y Manual de Usuario.

- **Manual de instalación:** contiene la descripción de las especificaciones mínimas para poder instalar la aplicación, además de información que permite que el sistema se ejecute según requerimientos de sistema tanto del lado del software como el de hardware.
- **Manual Técnico:** Está compuesto por artefactos producidos en el momento que se realizó la fase de análisis y diseño para el desarrollo del sistema. Contiene información sobre los requerimientos del sistema (modelo de casos de uso) y las interacciones de los usuarios con el sistema (diagramas de interacción), y la arquitectura que se manejó.
- **Manual de usuario:** Es una guía de apoyo que describe la forma en la que deben proceder los usuarios para hacer uso correcto de la aplicación y de las funcionalidades correspondientes a cada uno de los actores, permitiendo una óptima interacción del usuario con el software.

4. DESCRIPCIÓN DE LA ESTRUCTURA DE DIRECTORIOS, DESCRIPCIÓN DE LOS ARCHIVOS DE SOPORTE.

Muestra la distribución de las carpetas con los archivos necesarios para la operación del software por parte del cliente, servidor local y servidor esclavo, sobre una carpeta raíz que contiene todo lo necesario para el correcto funcionamiento de la aplicación.

Directorio Raíz:



Cliente



Esclavo



Local



esclavo.x



local.x

Directorio Cliente:



cliente



cliente.c



cliente.o



local.h



local.x



local_clnt.c



local_clnt.o



local_xdr.c

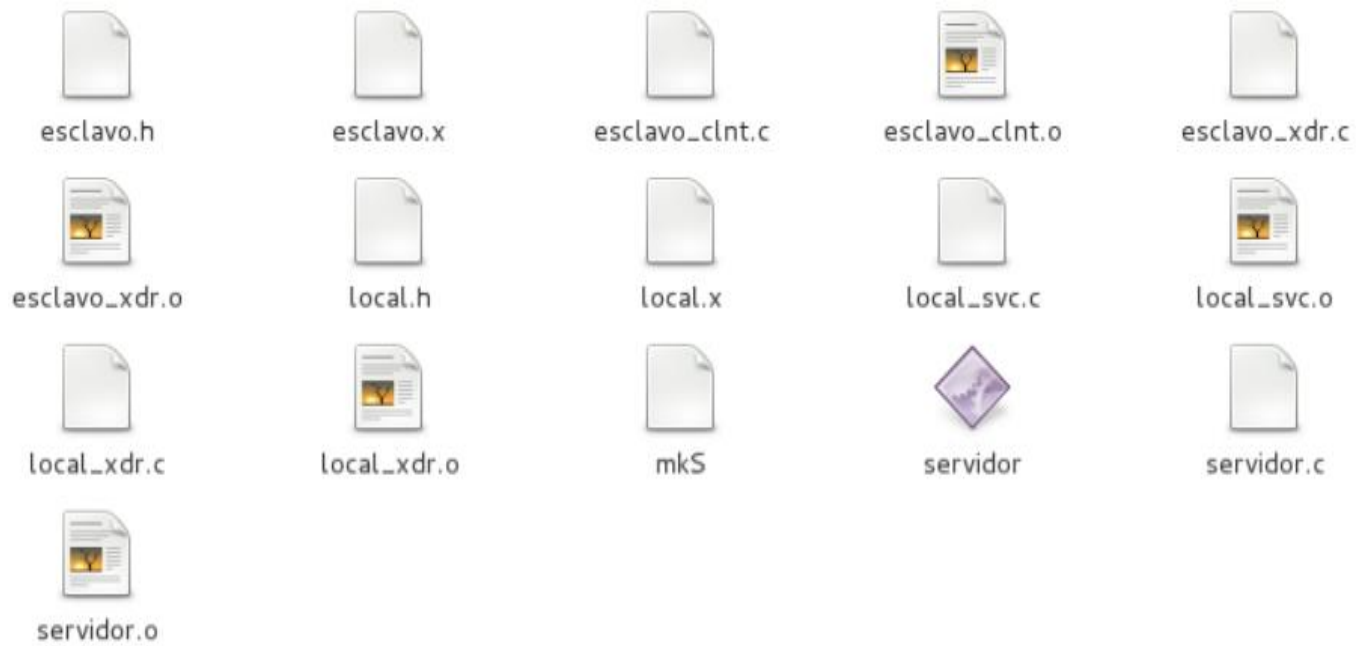


local_xdr.o



mkC

Directorio Local:



Directorio Esclavo:



5. DESCRIPCIÓN DE LOS ARCHIVOS FUENTE (INTERFACES, ARCHIVOS CLIENTE, ARCHIVOS SERVIDOR)

CLIENTE:

- local.x: Determina la interfaz en XDR, la cual permite generar las plantillas para el Stub del Cliente.
- local.h: Declarar los tipos de datos y las funciones.
- cliente.c: archivo con la lógica del cliente, se implementan las peticiones al servidor.
- local_xdr.c: archivo con la representación en lenguaje C de las estructuras en el archivo de interfaz XDR.
- local_clnt.c: archivo que soporta la funcionalidad de comunicación con el servidor para la aplicación. Archivo fuente del stub del Cliente.
- mkC: Corresponde al makefile del cliente.
- cliente: Ejecutable para la maquina en la que está el cliente.

LOCAL:

- servidor.c: archivo con la lógica del servidor, se implementan los servicios brindados por el sistema. Proviene de la interfaz local.x.
- local.x: Determina la interfaz en XDR, la cual permite generar las plantillas para el Stub del servidor (local).
- local.h: Declarar los tipos de datos y las funciones.
- local_xdr.c: archivo con la representación en lenguaje C de las estructuras en el archivo de interfaz XDR.
- local_svc.c: Archivo fuente del stub del servidor.
- esclavo.x: Determina la interfaz en XDR, la cual permite generar las plantillas para el Stub del Cliente.
- esclavo.h: Declarar los tipos de datos y las funciones.
- esclavo_xdr.c: archivo con la representación en lenguaje C de las estructuras en el archivo de interfaz XDR.

- esclavo_clnt.c: archivo que soporta la funcionalidad de comunicación con el servidor para la aplicación. Archivo fuente del stub del Cliente.
- mkS: Corresponde al makefile del servidor de la interfaz local.x.
- servidor: Ejecutable para la maquina en la que está el servidor de la interfaz local.x.

ESCLAVO:

- servidor.c: archivo con la lógica del servidor, se implementan los servicios brindados por el sistema. Proviene de la interfaz esclavo.x.
- esclavo.x: Determina la interfaz en XDR, la cual permite generar las plantillas para el Stub del servidor.
- esclavo.h: Declarar los tipos de datos y las funciones.
- esclavo_xdr.c: archivo con la representación en lenguaje C de las estructuras en el archivo de interfaz XDR.
- esclavo_svc.c: Archivo fuente del stub del servidor.
- mkS: Corresponde al makefile del servidor.
- servidor: Ejecutable para la maquina en la que está el servidor.

6. DESCRIPCIÓN DEL PROCESO DE COMPILACIÓN

Para poder compilar el programa se deben seguir los siguientes pasos:

Compilación de parte del cliente:

1. Se copia la carpeta Cliente, desde el directorio raíz, en la máquina en donde se va a correr el proceso cliente.
2. Se abre una terminal en la máquina cliente.
3. Se digita el comando para ejecutar el makefile del cliente y generar el ejecutable del cliente:

```
make -f mkC
```

Con lo cual se genera el ejecutable cliente.

4. Se termina con el proceso de compilación de parte del cliente.

Compilación de parte del servidor Local:

1. Se copia la carpeta Local, desde el directorio raíz, a la máquina en donde se va a correr el servidor local.
2. Se abre una terminal en la máquina donde se va a correr el servidor local.
3. Se digita el comando para ejecutar el makefile del servidor y generar el ejecutable del servidor:

```
make -f mkS
```

Con lo cual se genera el ejecutable servidor.

4. Se termina con el proceso de compilación de parte del servidor local.

Compilación de parte del servidor Esclavo:

1. Se copia la carpeta Esclavo, desde el directorio raíz, a las máquinas en donde se va a correr el servidor esclavo. Deben ser cinco máquinas.
2. Se abre una terminal en cada una de las máquinas donde se va a correr el servidor esclavo.
3. Se digita el comando para ejecutar el makefile del servidor y generar el ejecutable del servidor:

`make -f mkS`

Con lo cual se genera el ejecutable servidor.

4. Se termina con el proceso de compilación de parte del servidor esclavo.

7. DESCRIPCIÓN DEL PROCESO DE EJECUCIÓN

Para realizar este proceso, previamente se deben tener los ejecutables del punto anterior (Descripción del proceso de compilación).

Vale la pena recordar, que para el buen funcionamiento del sistema, el orden de ejecución es primero el servidor, quien al realizar dicha acción, se quedará esperando por las peticiones que el cliente envíe. Para este caso se debe ejecutar primero el servidor del proceso esclavo, luego el servidor del proceso local y finalmente el proceso cliente.

- Una vez creados los makeFile en cada carpeta (Cliente, Local y Esclavo) se procede a digitar lo siguiente:

Ejecución del lado del Esclavo:

Desde la terminal de la maquina donde se compiló el proceso esclavo ejecutaremos el servidor esclavo de la siguiente manera:

```
. /servidor
```

Ejecución del lado del Local:

Desde la terminal de la maquina donde se compiló el proceso local ejecutaremos el servidor esclavo de la siguiente manera:

```
. /servidor
```

Ejecución del lado del Cliente:

En todas las maquinas donde se vaya a lanzar el proceso cliente, se debe ejecutar el archivo cliente de la siguiente manera:

```
. /cliente 'ip máquina del servidor'
```

Para obtener el ip de la maquina donde está ubicado el servidor, se ejecuta el siguiente comando sobre el bash:

```
ifconfig
```

El cual nos dará el ip de la maquina servidor y será el que se ingresa a la hora de ejecutar el cliente.

Si el proceso esclavo se va ejecutar en la misma máquina donde se va a correr el proceso servidor local y servidor esclavo el comando es:

```
./cliente localhost
```