



Facultad de Ingeniería

Escuela de Ingeniería en Bioinformática

---

# Reporte:

# Administración de Sistemas

---

*Alumno:*

Andrés Nicolás Cofré Díaz

Matrícula:

2014430845

*Profesor:*

Alejandro Mauricio Valdes Jimenez

# *Contents*

<b>1</b>	<b>Lección 1: Compilando programas, librerías, kernel</b>	<b>8</b>
1.1	Contexto General . . . . .	8
1.2	GCC . . . . .	9
1.2.1	Trabajo Realizado . . . . .	9
1.2.1.1	Compilación Librería GMP . . . . .	9
1.2.1.2	Compilación Librería MPFR . . . . .	12
1.2.1.3	Compilación Librería MPC . . . . .	12
1.2.1.4	Compilación Librería ISL . . . . .	13
1.2.1.5	Compilador GCC 7.5.0 . . . . .	15
1.3	KDTREE . . . . .	17
1.3.1	Trabaja Realizado . . . . .	17
1.3.2	Ejemplo . . . . .	18
1.4	Kernel . . . . .	19
1.4.1	Trabajo Realizados . . . . .	19

<b>2</b>	<b>Lección 2: Administración de servicios - Acceso Remoto</b>	<b>24</b>
2.1	Contexto General . . . . .	24
2.2	Telnet . . . . .	24
2.2.1	Trabajo Realizado . . . . .	24
2.3	SSH . . . . .	28
2.3.1	Trabajo Realizado . . . . .	28
2.3.1.1	SCP . . . . .	28
2.3.1.2	SSHFS . . . . .	29
2.3.1.3	Relación de Confianza . . . . .	32
<b>3</b>	<b>Lección 3: Administración de servicios-Transferencia de Datos</b>	<b>35</b>
3.1	Contexto General . . . . .	35
3.2	Práctica: TFTP . . . . .	35
3.3	Práctica: FTP . . . . .	35
<b>4</b>	<b>Problemas Encontrados</b>	<b>36</b>
4.1	Lección 1 . . . . .	36
4.1.1	Practica: Compilación de GCC . . . . .	36
4.1.2	Practica: Kdtree . . . . .	36
4.1.3	Practica: Kernel . . . . .	36
4.2	Lección 2 . . . . .	37
4.2.1	Practica: Telnet . . . . .	37

4.2.2	Practica: SSH . . . . .	37
4.3	Lección 3 . . . . .	37
4.3.1	Practica: TFTP . . . . .	37
4.3.2	Practica: FTP . . . . .	37

## *List of Figures*

1.1	Versión instalada por defecto . . . . .	9
1.2	Descarga de la librería GMP . . . . .	9
1.3	Descompresión de la librería . . . . .	10
1.4	Configuración de la librería GMP . . . . .	10
1.5	Make de la librería GMP . . . . .	11
1.6	Make Check de la librería GMP . . . . .	11
1.7	Resumen del Make Check . . . . .	11
1.8	Obtención y descompresión de la librería MPC . . . . .	13
1.9	Make Install de la Librería MPC . . . . .	13
1.10	Obtención y bunzip del comprimido ISL . . . . .	14
1.11	Configuración de la librería ISL . . . . .	14
1.12	Obtención de la versión del GCC . . . . .	15
1.13	Antes y Después de la creación de la carpeta . . . . .	15
1.14	Configuración del GCC . . . . .	16
1.15	Instalación terminada del GCC . . . . .	17

1.16 Obtención de kdtree . . . . .	17
1.17 Configuración de Kdtree . . . . .	18
1.18 Realización del Make de kdtree. . . . .	18
1.19 Ejemplo ejecutado. . . . .	19
1.20 Instalación de las herramientas. . . . .	20
1.21 Antes de la instalación del kernel. . . . .	20
1.22 Limpieza de las configuraciones previas. . . . .	21
1.23 Error en la compilación. . . . .	21
1.24 Archivos generados en la compilación. . . . .	22
1.25 Instalación del Kernel. . . . .	22
1.26 Kernel instalado correctamente. . . . .	23
2.1 Instalación de Telnet. . . . .	25
2.2 Estado de Telnet. . . . .	25
2.3 Antes de la creación del usuario. . . . .	26
2.4 Creación del usuario con su contraseña. . . . .	26
2.5 Despues de crear el usuario. . . . .	27
2.6 Conexión Realizada. . . . .	27
2.7 Conexiones Activas. . . . .	28
2.8 Comprobar contenido previo a la transferencia. . . . .	28
2.9 Transferencia de un archivo al servidor remoto. . . . .	29

2.10 Comprobar la transferencia. . . . .	29
2.11 Instalación de SSHFS. . . . .	30
2.12 Creación de la carpeta en el host remoto. . . . .	30
2.13 Contenido antes de realizar el anclaje. . . . .	31
2.14 Se ancla el home del host remoto. . . . .	31
2.15 Después de anclar. . . . .	32
2.16 Generación de la Key. . . . .	33
2.17 Almacenaje de las key y asignación de las claves. . . . .	33
2.18 Se salta la ultima contraseña. . . . .	34
2.19 Almacenaje de las key y asignación de las claves para la otra conexión . . . .	34
2.20 Relación de Confianza creada . . . . .	34

# 1. *Lección 1: Compilando programas, librerías, kernel*

## 1.1 Contexto General

En esta parte se requiere que se compile una diferentes herramientas como es el caso del GCC (Ver 1.2), el cual corresponde a un compilador integrado para los lenguajes de programación C,C++ y Fortrain, cuyas siglas corresponden a GNU Compiler Collection. Esta compilación se realizara para probar ciertos códigos. Esto se debe a que existen ciertas interacciones que no se pueden realizar con la versión del compilador con la que viene Linux instalado por defecto en el dispositivo, se desea aprender a instalar una nueva versión del GCC, y de otras herramientas como lo es kdtree (Ver 1.3) el cual corresponde a un árbol de búsqueda binaria en donde los datos organizados en nodos poseen una dimensión K, utilizada también como librería en el lenguaje de programación C y debido a la misma razón que para el compilador al no existir en ciertas versiones se debe adquirir. Además de realizar esto en el dispositivo personal se empleara uso de un servidor, para hacer una compilación de mayor impacto como lo es el Kernel (Ver 1.4 ), el cual es el componente central de un sistema operativo el cual se encarga de gestionar las operaciones del computador y del hardware; esto se realiza debido a que cada versión que se lanza posee características nuevas y algunas de las que vienen en la anterior puede que ya no esten disponibles debido a que ya no son requeridas.



## 1.2 GCC

### 1.2.1 Trabajo Realizado

Antes de realizar la compilación del nuevo GCC, se debe compilar primero una serie de librerías necesarias. Al poseer la versión 6.3.0, se decidió por hacer la compilación de la versión 7.5.0.

```
(base) andres@Andres:~$ gcc --version
gcc (Debian 6.3.0-18+deb9u1) 6.3.0 20170516
Copyright (C) 2016 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

Figure 1.1: Versión instalada por defecto

#### 1.2.1.1 Compilación Librería GMP

Se comenzó por la librería GMP en su ultima versión la cual corresponde a la 6.1.0, por lo que utilizando el comando **wget ftp://gcc.gnu.org/pub/gcc/infrastructure/gmp-6.1.0.tar.bz2** se descargo el archivo en la carpeta que se esta trabajando.

```
(base) andres@Andres:~/Escritorio/Adm. Sistemas$ cd Actividad1/
(base) andres@Andres:~/Escritorio/Adm. Sistemas/Actividad1$ wget ftp://gcc.gnu.org/pub/gcc/infrastructure/gmp-6.1.0.tar.bz2
--2020-09-24 09:03:36--  ftp://gcc.gnu.org/pub/gcc/infrastructure/gmp-6.1.0.tar.bz2
=> "gmp-6.1.0.tar.bz2"
Resolviendo gcc.gnu.org (gcc.gnu.org)... 8.43.85.97, 2620:52:3:1:0:246e:9693:128c
Conectando con gcc.gnu.org (gcc.gnu.org)[8.43.85.97]:21... conectado.
Identificándose como anonymous ... ¡Dentro!
==> SYST ... hecho. ==> PWD ... hecho.
==> TYPE I ... hecho. ==> CWD (1) /pub/gcc/infrastructure ... hecho.
==> SIZE gmp-6.1.0.tar.bz2 ... 2383840
==> PASV ... hecho. ==> RETR gmp-6.1.0.tar.bz2 ... hecho.
Longitud: 2383840 (2,3M) (probablemente)

gmp-6.1.0.tar.bz2 100%[=====] 2,27M 1,85MB/s in 1,2s
2020-09-24 09:03:39 (1,85 MB/s) - "gmp-6.1.0.tar.bz2" guardado [2383840]
```

Figure 1.2: Descarga de la librería GMP

Como se puede ver el archivo que se descargo esta comprimido en el formato tar.bz2,

en donde la extensión bz2 o bunzip2 es utilizada solo para los sistemas basados en Unix para distribución de software, a menudo utilizada para los contenedores de archivos que no soportan por ejemplo la compresión como los del tipo TAR. Para poder descomprimirlo se debe utilizar el comando **bunzip2 gmp-6.1.0.tar.bz2** luego de esto el archivo quedara en el formato .tar por lo que se debe descomprimir nuevamente, esta vez con el comando **tar xvf gmp-6.1.0.tar**.

```
(base) andres@Andres:~/Escritorio/Adm. Sistemas/Actividad1$ bunzip2 gmp-6.1.0.tar.bz2
(base) andres@Andres:~/Escritorio/Adm. Sistemas/Actividad1$ tar xvf gmp-6.1.0.tar
```

Figure 1.3: Descompresión de la librería

Luego de descomprimir los archivos se ingreso a la carpeta generada con **cd nombre-delacarptay** se utilizo el comando config para realizar la configuración de esta librería, el comando completo que se utiliza para este caso es **./configure --disable-shared --enable-static --prefix=/opt/util\_compiler/**.

```
(base) andres@Andres:~/Escritorio/Adm. Sistemas/Actividad1/gmp-6.1.0$ ./configure --disable-shared --enable-static --prefix=/opt/util_compiler/
checking build system type... skylake-pc-linux-gnu
checking host system type... skylake-pc-linux-gnu
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
checking whether to enable maintainer-specific portions of Makefiles... no
checking ABI=64
checking compiler gcc -O2 -pedantic -fomit-frame-pointer -m64 ... yes
checking compiler gcc -O2 -pedantic -fomit-frame-pointer -m64 -mtune=skylake... yes
checking compiler gcc -O2 -pedantic -fomit-frame-pointer -m64 -mtune=skylake -march=broadwell... yes
```

Figure 1.4: Configuración de la librería GMP

Una vez finalizada la configuración se realiza un make el cual se utiliza determinar automáticamente que partes de un programa largo deben ser recompiladas y un make check para realizar lo que esta definido por el Makefile, en donde en ciertos puntos el make check arrojará resúmenes de lo que fue realizando, un ejemplo de esos resúmenes se puede ver en la siguiente imagen.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0$ make
make all-recursive
make[1]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0'
Making all in tests
make[2]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests'
Making all in .
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests'
make[3]: No se hace nada para 'all-am'.
make[3]: se sale del directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests'
Making all in devel
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/devel'
make[3]: No se hace nada para 'all'.
make[3]: se sale del directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/devel'
Making all in mpn
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpn'
make[3]: No se hace nada para 'all'.
make[3]: se sale del directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpn'
Making all in mpz
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpz'
make[3]: No se hace nada para 'all'.
make[3]: se sale del directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpz'
Making all in mpq
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpq'
make[3]: No se hace nada para 'all'.
make[3]: se sale del directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpq'
Making all in mpf
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpf'
make[3]: No se hace nada para 'all'.
make[3]: se sale del directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/mpf'
Making all in rand
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/rand'
make[3]: No se hace nada para 'all'.
make[3]: se sale del directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/rand'
Making all in misc
make[3]: se entra en el directorio '/home/andres/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0/tests/misc'
make[3]: No se hace nada para 'all'.
```

Figure 1.5: Make de la librería GMP

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/gmp-6.1.0$ make check
```

Figure 1.6: Make Check de la librería GMP

```
=====
Testsuite summary for GNU MP 6.1.0
=====
# TOTAL: 8
# PASS: 8
# SKIP: 0
# XFAIL: 0
# FAIL: 0
# XPASS: 0
# ERROR: 0
=====
```

Figure 1.7: Resumen del Make Check

Para finalizar con esta librería se debe estar como por superusuario y realizar el **make install** y una vez que finalice se sale del modo superusuario con el comando **exit** y regresamos a la carpeta anterior con **cd ...**

### 1.2.1.2 Compilación Librería MPFR

Una vez que se terminara con la librería GMP se continuo con la MPFR, en su ultima versión , la cual se obtiene con el comando **wget ftp://gcc.gnu.org/pub/gcc/infrastructure/mpfr-3.1.4.tar.bz2**. El procedimiento que se le sigue similar a la utilizada para la librería anterior en cuanto a descomprimir y realizar los makes, por lo que se utiliza bunzip2 y tar xvf para descomprimir el archivo, al momento de descomprimirlo entramos a esa carpeta con **cd mpfr-3.1.4**. En donde se diferencia es en la configuración, dado que utiliza aparte de su propio prefijo, el prefijo que se define en la configuración anterior, el comando completo que se utiliza es el **./configure --disable-shared --enable-static --prefix=/opt/util\_compiler --with-gmp=/opt/util\_compiler** y de igual manera que se realizo con la librería GMP se hace un make y make check los cuales se pueden correr en la misma linea, ingresándolos de la siguiente manera **make && make install**, luego de esto como superusuario se realiza el **make install**, al finalizar, como se hizo antes hay que salir del como superusuario y regresar a la carpeta anterior.

### 1.2.1.3 Compilación Librería MPC

Ahora se debe obtener el la librería MPC con **wget ftp://gcc.gnu.org/pub/gcc/infrastructure/mpc-1.0.3.tar.gz** , en este caso el archivo que se obtiene es un \*.tar.gz por lo que para descomprimirlo se debe usar **tar zxvf mpc-1.0.3.tar.gz**. Una vez que se descomprime con el uso del comando cd ingresamos en esta carpeta.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ wget ftp://gcc.gnu.org/pub/gcc/infrastructure/mpc-1.0.3.tar.gz
--2020-09-24 12:50:27-- ftp://gcc.gnu.org/pub/gcc/infrastructure/mpc-1.0.3.tar.gz
=> "mpc-1.0.3.tar.gz"
Resolviendo gcc.gnu.org (gcc.gnu.org)... 8.43.85.97, 2620:52:3:1:0:246e:9693:128c
Conectando con gcc.gnu.org (gcc.gnu.org)[8.43.85.97]:21... conectado.
Identificándose como anonymous ... ¡Dentro!
==> SYST ... hecho. ==> PWD ... hecho.
==> TYPE I ... hecho. ==> CWD (1) /pub/gcc/infrastructure ... hecho.
==> SIZE mpc-1.0.3.tar.gz ... 669925
==> PASV ... hecho. ==> RETR mpc-1.0.3.tar.gz ... hecho.
Longitud: 669925 (654K) (probablemente)

mpc-1.0.3.tar.gz          100%[=====] 654,22K  928KB/s  in
0,7s

2020-09-24 12:50:29 (928 KB/s) - "mpc-1.0.3.tar.gz" guardado [669925]

(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ tar zxvf mpc-1.0.3.tar.gz
```

Figure 1.8: Obtención y descompresión de la librería MPC

En el caso de la configuración de esta librería, se utiliza los prefijos definidos en las configuraciones anteriores por lo que el comando a utilizar es el siguiente: **./configure --disable-shared --enable-static --prefix=/opt/util\_compiler --with-gmp=/opt/util\_compiler --with-mpfr=/opt/util\_compiler**. Ahora se debe realizar los makes se debe asignar cpu, en el caso de este dispositivo, el cual consta de 4 de CPU se asignan 2 para la realización del make, para esto se utiliza el comando: **make -j 2 && make check** en caso de contar con una mayor cantidad de CPU se puede reemplazar la cantidad que se menciona, para poder saber cuanta CPU se posee utilizar el comando **top** o **htop**, las CPU aparecerán en negrita. Finalmente al igual que en los casos anteriores se debe realizar un **make install** como superusuario. Al finalizar se regresa a la carpeta anterior y se sale del modo superusuario.

```
-----
root@Andres:/home/andres/Escritorio/Adm.Sistemas/Actividad1/mpc-1.0.3# make install
```

Figure 1.9: Make Install de la Librería MPC

#### 1.2.1.4 Compilación Librería ISL

Finalmente la librería ISL se obtiene con **wget ftp://gcc.gnu.org/pub/gcc/infrastructure/isl-0.16.1.tar.bz2**, y al ser este un .bz2 se debe descomprimir utilizando **bunzip2 isl-0.16.1.tar.bz2**.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ wget ftp://gcc.gnu.org/pub/gcc/infrastructure/isl-0.16.1.tar.bz2
--2020-09-24 14:40:29-- ftp://gcc.gnu.org/pub/gcc/infrastructure/isl-0.16.1.tar.bz2
=> "isl-0.16.1.tar.bz2"
Resolviendo gcc.gnu.org (gcc.gnu.org)... 8.43.85.97, 2620:52:3:1:0:246e:9693:128c
Conectando con gcc.gnu.org (gcc.gnu.org)[8.43.85.97]:21... conectado.
Identificándose como anonymous ... ¡Dentro!
==> SYST ... hecho. ==> PWD ... hecho.
==> TYPE I ... hecho. ==> CWD (1) /pub/gcc/infrastructure ... hecho.
==> SIZE isl-0.16.1.tar.bz2 ... 1626446
==> PASV ... hecho. ==> RETR isl-0.16.1.tar.bz2 ... hecho.
Longitud: 1626446 (1,6M) (probablemente)

isl-0.16.1.tar.bz2          100%[=====>]  1,55M  1,55MB/s   in 1,0s

2020-09-24 14:40:33 (1,55 MB/s) - "isl-0.16.1.tar.bz2" guardado [1626446]

(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ bunzip2
gmp-6.1.0/      isl-0.16.1.tar.bz2  mpc-1.0.3/      mpfr-3.1.4/
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ bunzip2 isl-0.16.1.tar.bz2
```

Figure 1.10: Obtención y bunzip del comprimido ISL

Una vez que se realizó el bunzip se debe descomprimir con **tar xvf isl-0.16.1.tar**, luego de esto ingresamos en la carpeta generada. Para la configuración a diferencia del caso anterior en el cual se utilizaban los prefijos definidos anteriormente, acá solo se configura utilizando el definido para la librería GMP, por lo que el comando a utilizar es el siguiente:

**./configure --with-gmp-prefix=/opt/util\_compiler --disable-shared --enable-static --prefix=/opt/util\_compiler**

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ cd isl-0.16.1/
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/isl-0.16.1$ ./configure --with-gmp-prefix=/opt/util_compiler --disable-shared --enable-static --prefix=/opt/util_compiler
```

Figure 1.11: Configuración de la librería ISL

Finalmente como se hizo con la librería anterior al momento de realizar el make y make check se asignaron 2 de CPU, por lo que el comando que se utiliza es el mismo, es decir: **make -j 2 && make check**, luego al igual que en todos los casos anteriores como superusuario se realiza el **make install**.

### 1.2.1.5 Compilador GCC 7.5.0

Utilizando la pagina <https://ftp.gnu.org/gnu/gcc/> se busco la versión que se desea instalar, en este caso la 7.5.0, por lo que una vez encontrada esa versión se ingresa en esa carpeta y copia la dirección del .tar.gz para obtenerlo a través de wget, por lo que el comando completo que se utiliza pa este caso es: **wget https://ftp.gnu.org/gnu/gcc/gcc-7.5.0/gcc-7.5.0.tar.gz**.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ wget https://ftp.gnu.org/gnu/gcc/gcc-7.5.0/gcc-7.5.0.tar.gz
--2020-09-24 15:11:59-- https://ftp.gnu.org/gnu/gcc/gcc-7.5.0/gcc-7.5.0.tar.gz
Resolviendo ftp.gnu.org (ftp.gnu.org)... 209.51.188.20, 2001:470:142:3::b
Conectando con ftp.gnu.org (ftp.gnu.org)[209.51.188.20]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 112743638 (108M) [application/x-gzip]
Grabando a: "gcc-7.5.0.tar.gz"

gcc-7.5.0.tar.gz          3%[=>]          3,51M  122KB/s   eta 19m 7s
```

Figure 1.12: Obtención de la versión del GCC

Una vez obtenido se continua con la descompresión de la carpeta. Una vez que se descomprime se debe crear una nueva carpeta llamada build\_gcc con el comando **mkdir build\_gcc**

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ ls
gcc-7.5.0  gcc-7.5.0.tar.gz  gmp-6.1.0  gmp-6.1.0.tar  isl-0.16.1  isl-0.16.1.tar  mpc-1.0.3  mpc-1.0.3.tar.gz  mpfr-3.1.4  mpfr-3.1.4.tar
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ mkdir build_gcc
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$ ls
build_gcc  gcc-7.5.0.tar.gz  gmp-6.1.0.tar  isl-0.16.1.tar  mpc-1.0.3.tar.gz  mpfr-3.1.4.tar
gcc-7.5.0  gmp-6.1.0        isl-0.16.1    mpc-1.0.3      mpfr-3.1.4
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1$
```

Figure 1.13: Antes y Después de la creación de la carpeta

una vez que se crea la carpeta, ingresamos en esta y desde aquí se hace la configuración del GCC, para este caso se utiliza este comando: **./gcc-7.5.0/configure --enable-shared --enable-threads=posix --enable-\_\_cxa\_atexit --enable-clocale=gnu --enable-languages=c,c++,for --prefix=/opt/gcc/v7.5.0 --disable-multilib --with-gmp=/opt/util\_compiler --with-mpc=/opt/util\_compiler --with-mpfr=/opt/util\_compiler --with-isl=/opt/util\_compiler**, en el caso de estar utilizando para otra versión se debe reemplazar 7.5.0 por la que se desea

configurar. Luego de esto se debe realizar un **make -j 2**, en donde el 2 corresponde a la CPU que utilizara este proceso, y estar atento si es que aparece algún error. Hay que señalar que este proceso demora, en este caso tardo 2 horas en finalizar sin errores. Una vez que finalice sin encontrar errores, como superusuario se realiza un **make install-strip** el cual tarda un poco menos que el procedimiento anterior.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/build_gcc$ ../gcc-7.5.0/configure --enable-shared --enable-threads=posix --enable-
_cxa_atexit \
> --enable-clocale=gnu --enable-languages=c,c++,fortran,objc \
> --prefix=/opt/gcc/v7.5.0 --disable-multilib \
> --with-gmp=/opt/util_compiler \
> --with-mpc=/opt/util_compiler \
> --with-mpfr=/opt/util_compiler \
> --with-isl=/opt/util_compiler[]
```

Figure 1.14: Configuración del GCC



Una vez que finalice si se quiere saber si se instalo bien como superusuario dentro de la carpeta gcc\_build realizar un `./gcc --version`

```
root@Andres:/home/andres/Escritorio/Adm.Sistemas/Actividad1/gcc-7.5.0# cd /opt/gcc/v7.5.0/bin/
root@Andres:/opt/gcc/v7.5.0/bin# ./gcc --version
gcc (GCC) 7.5.0
Copyright (C) 2017 Free Software Foundation, Inc.
Esto es software libre; vea el código para las condiciones de copia. NO hay
garantía; ni siquiera para MERCANTIBILIDAD o IDONEIDAD PARA UN PROPÓSITO EN
PARTICULAR
```

Figure 1.15: Instalación terminada del GCC

## 1.3 KDTREE

### 1.3.1 Trabaja Realizado

Se comienza accediendo a la pagina <http://nuclear.mutantstargoat.com/sw/kdtree/> para poder obtener la ultima versión de kdtree, la cual en este caso es la 0.5.7. Para poder realizar la descarga, se utilizo wget para poder obtener la carpeta de dicha versión; siendo el comando completo: `wget http://nuclear.mutantstargoat.com/sw/kdtree/files/kdtree-0.5.7.tar.gz`.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/Actividad1B$ wget http://nuclear.mutantstargoat.com/sw/kdtree/files/kdtree-0.5.7.tar.gz
--2020-10-09 15:15:29-- http://nuclear.mutantstargoat.com/sw/kdtree/files/kdtree-0.5.7.tar.gz
Resolviendo nuclear.mutantstargoat.com (nuclear.mutantstargoat.com)... 176.28.20.253
Conectando con nuclear.mutantstargoat.com (nuclear.mutantstargoat.com)[176.28.20.253]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 11215 (11K) [application/x-gzip]
Grabando a: "kdtree-0.5.7.tar.gz"

kdtree-0.5.7.tar.gz          100%[=====] 10,95K  --.-KB/s  in 0s
2020-10-09 15:15:30 (72,7 MB/s) - "kdtree-0.5.7.tar.gz" guardado [11215/11215]
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/Actividad1B$ █
```

Figure 1.16: Obtención de kdtree

Una vez descargado y descomprimido el contenedor se accedió a este a través de la

terminal y se realiza una configuración, utilizando `./configure` para la preparación del make file, esto se ve en la siguiente imagen.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/Actividad1B/kdtree-0.5.7$ ./configure
configuring kdtree ...
prefix: /usr/local
optimize for speed: yes
include debugging symbols: yes
fast node allocator: yes
pthread support: yes
creating makefile ...
configuration completed, type make (or gmake) to build.
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/Actividad1B/kdtree-0.5.7$
```

Figure 1.17: Configuración de Kdtree

Una vez que la configuración finalice se realiza un **make** para terminar la preparación. Cabe recalcar que al no desear instalar kdtree en el dispositivo no hay necesidad de hacer un make install como superusuario.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/Actividad1B/kdtree-0.5.7$ make
cc -pedantic -Wall -fPIC -O3 -g -DUSE_LIST_NODE_ALLOCATOR -c -o kdtree.o kdtree.c
ar rcs libkdtree.a kdtree.o
cc -shared -Wl,-soname=libkdtree.so.0 -o libkdtree.so.0.1 kdtree.o -lpthread
[ -n "libkdtree.so.0" ] && ln -s libkdtree.so.0.1 libkdtree.so.0 || true
[ -n "libkdtree.so" ] && ln -s libkdtree.so.0 libkdtree.so || true
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/Actividad1B/kdtree-0.5.7$ █
```

Figure 1.18: Realización del Make de kdtree.

### 1.3.2 Ejemplo

Se ubico en la carpeta examples de kdtree y se realizo un make para que se generen los ejecutables, una vez hecho esto se ejecuto el ejemplo 2 con el comando `./test2`, lo que genero es un numero aleatorio de nodos, de 3 dimensiones. Este código se modifico para de esta manera poder conocer de que manera se estan realizando los cálculos.

```
(base) andres@Andres:~/Escritorio/Adm.Sistemas/Actividad1/Actividad1B/kdtree-0.5.7/examples$ ./test2
found 3 results:
a3 1.000 b3 0.072 diff: 0.928 y diff*diff; 0.862 dist_sq: 0.862
a2 0.000 b2 -4.861 diff: 4.861 y diff*diff; 23.633 dist_sq: 24.494
a1 0.000 b1 3.659 diff: -3.659 y diff*diff; 13.386 dist_sq: 37.880
node at (3.659, -4.861, 0.072) is 6.155 away and has data=h
a3 1.000 b3 4.838 diff: -3.838 y diff*diff; 14.727 dist_sq: 14.727
a2 0.000 b2 1.030 diff: -1.030 y diff*diff; 1.061 dist_sq: 15.788
a1 0.000 b1 -1.626 diff: 1.626 y diff*diff; 2.644 dist_sq: 18.432
node at (-1.626, 1.030, 4.838) is 4.293 away and has data=f
a3 1.000 b3 -5.012 diff: 6.012 y diff*diff; 36.147 dist_sq: 36.147
a2 0.000 b2 1.836 diff: -1.836 y diff*diff; 3.372 dist_sq: 39.520
a1 0.000 b1 5.464 diff: -5.464 y diff*diff; 29.858 dist_sq: 69.377
node at (5.464, 1.836, -5.012) is 8.329 away and has data=c
```

Figure 1.19: Ejemplo ejecutado.

Aquí el  $a$  corresponde al vector  $(0, 0, 1)$  y  $b$  a las coordenadas de los nodos,  $\text{diff}$  es la diferencia entre el vector y nodo, y en el caso de  $\text{dist\_sq}$  es la sumatoria de las distancia al cuadrado; esto quiere decir que al calcular  $\text{diff}$  este sera elevado al cuadrado y sumado con  $\text{dist\_sq}$ . Finalmente para sacar su distancia al finalizar el calculo se saca la raíz de  $\text{dist\_sq}$ . Hay que señalar que la manera en que opera este método es a la inversa, esto quiere decir que en vez de partir con el primer dato del vector y el nodo, se inicia con el ultimo además de que solo se resta cuando el valor de  $i=j$ , debido a que las variables estan definidas como  $a_i$  y  $b_j$ . Ahora la manera en que se determina el dato mas cercano es asignándole a cada coordenada una letra del abecedario y evaluando a cual de estos esta más cerca gracias a la distancia calculada anteriormente.

## 1.4 Kernel

### 1.4.1 Trabajo Realizados

Se comienza por la conexión al servidor a traves del comando **ssh** el cual pedira dos claves dado que son 2 conexiones simultaneas en donde la segunda es para conectarse a la maquina virtual asignada. Una vez conectado se abre el archivo `/etc/apt/sources.list` y se reemplaza todo lo que diga `debian.utalca.cl` por `ftp.cl.debian.org`, debido a problemas con el servidor,

luego de esto se realiza un **apt uptade**. Hecho el uptade se instalan algunas herramientas utilizando **apt install build-essential** .

```

root@admSist2:/home/alumno# apt install build-essential
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  dirmngr dpkg-dev fakeroot gnupg gnupg-l10n gnupg-utils gpg gpg-agent gpg-wks-client gpg-wks-server gpgconf gpgsm libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libassuan0 libdpkg-perl libfakeroot libfile-fcntllock-perl libksba8 libnpt0 make patch
  pinentry-curses
Suggested packages:
  dbus-user-session pinentry-gnome3 tor debian-keyring parcimonie xloadimage sddm git bzr make-doc ed diffutils-doc pinentry-doc
The following NEW packages will be installed:
  build-essential dirmngr dpkg-dev fakeroot gnupg gnupg-l10n gnupg-utils gpg gpg-agent gpg-wks-client gpg-wks-server gpgconf gpgsm
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libassuan0 libdpkg-perl libfakeroot libfile-fcntllock-perl libksba8
  libnpt0 make patch pinentry-curses
0 upgraded, 25 newly installed, 0 to remove and 5 not upgraded.
Need to get 10.9 MB/11.0 MB of archives.
After this operation, 21.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ftp.cl.debian.org/debian buster/main amd64 make amd64 4.2.1-1.2 [341 kB]
Get:2 http://ftp.cl.debian.org/debian buster/main amd64 libdpkg-perl all 1.19.7 [1,414 kB]
Get:3 http://ftp.cl.debian.org/debian buster/main amd64 dpkg-dev all 1.19.7 [1,773 kB]
Get:4 http://ftp.cl.debian.org/debian buster/main amd64 build-essential amd64 12.6 [7,576 B]
Get:5 http://ftp.cl.debian.org/debian buster/main amd64 libassuan0 amd64 2.5.2-1 [49.4 kB]
Get:6 http://ftp.cl.debian.org/debian buster/main amd64 gpgconf amd64 2.2.12-1+deb10u1 [510 kB]
Get:7 http://ftp.cl.debian.org/debian buster/main amd64 libksba8 amd64 1.3.5-2 [99.7 kB]
Get:8 http://ftp.cl.debian.org/debian buster/main amd64 libnpt0 amd64 1.6-1 [18.4 kB]
Get:9 http://ftp.cl.debian.org/debian buster/main amd64 dirmngr amd64 2.2.12-1+deb10u1 [712 kB]
Get:10 http://ftp.cl.debian.org/debian buster/main amd64 libfakeroot amd64 1.23-1 [45.9 kB]
Get:11 http://ftp.cl.debian.org/debian buster/main amd64 fakeroot amd64 1.23-1 [85.8 kB]
Get:12 http://ftp.cl.debian.org/debian buster/main amd64 gnupg-l10n all 2.2.12-1+deb10u1 [1,010 kB]
Get:13 http://ftp.cl.debian.org/debian buster/main amd64 gnupg-utils amd64 2.2.12-1+deb10u1 [861 kB]
Get:14 http://ftp.cl.debian.org/debian buster/main amd64 gpg amd64 2.2.12-1+deb10u1 [865 kB]
Get:15 http://ftp.cl.debian.org/debian buster/main amd64 pinentry-curses amd64 1.1.0-2 [64.5 kB]
Get:16 http://ftp.cl.debian.org/debian buster/main amd64 gpg-agent amd64 2.2.12-1+deb10u1 [617 kB]

```

Figure 1.20: Instalación de las herramientas.

Una vez instaladas se descarga el kernel, el cual en este caso es la versión 5.2.11, utilizando **wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.2.11.tar.xz**. Antes de descomprimir el archivo, se revisa la versión actualmente instalada del kernel con **uname -r**. La siguiente imagen muestra la versión que se posee en un inicio.

```

root@admSist2:/usr/src# uname -r
4.19.0-10-amd64

```

Figure 1.21: Antes de la instalación del kernel.

Ahora como superusuario; dado que se requieren los permisos, se copia la comprimido en la dirección **/usr/src/** para luego ser descomprimido con **tar axvf linux-5.2.11.tar.xz** y una vez descomprimido se accede a la carpeta creada. Aquí se limpiarán las configuraciones

que habian anteriormente a través de **make distclean && make clean** tal cual lo muestra la siguiente imagen.

```
root@admSist2:/usr/src/linux-5.2.11# make distclean
root@admSist2:/usr/src/linux-5.2.11# make clean
root@admSist2:/usr/src/linux-5.2.11# █
```

Figure 1.22: Limpieza de las configuraciones previas.

Ahora se copia la configuración que posee el kernel instalado actualmente a través de **cp /boot/config-4.19.0-5-amd64 .config** para luego configurar las opciones del kernel con **make oldconfig**, pero este ultimo comando falla en una primera instancia debido a que no se estaba instalado flex, el que se instalo para volver a correr el comando anterior. Ahora se revisa que características nuevas se desean activar.

Ahora con el editor de texto vi o vim se refisa el archivo .config de la siguiente manera **vi .config** en el se deshabita el debug del kernel por lo que se utiliza el comando de búsqueda **/CONFIG\_DEBUG\_INFO=**, en donde se debe igualar a n. En el mismo archivo se busca de la misma manera **CONFIG\_SYSTEM\_TRUSTED\_KEYS=** y se borra todo lo que este entre las comillas sin borrar estas.

Ahora se realiza la compilación a través de **make deb-pkg** pero como es la primera vez que se compila el kernel arroja el siguiente error el cual indica que faltan librerías por instalar.

```
root@admSist2:/usr/src# dpkg -i linux-image-5.2.11_5.2.11-1_amd64.deb
dpkg: warning: 'ldconfig' not found in PATH or not executable
dpkg: warning: 'start-stop-daemon' not found in PATH or not executable
dpkg: error: 2 expected programs not found in PATH or not executable
Note: root's PATH should usually contain /usr/local/sbin, /usr/sbin and /sbin
root@admSist2:/usr/src# █
```

Figure 1.23: Error en la compilación.

Luego de realizar la instalación de las librerías y de compilar el kernel se generan los siguientes archivos que nos indicara que la compilación termino correctamente.

```
root@admSist2:/usr/src# ls
linux-5.2.11
linux-5.2.11_5.2.11-1_amd64.buildinfo
linux-5.2.11_5.2.11-1_amd64.changes
linux-5.2.11_5.2.11-1.diff.gz
linux-5.2.11_5.2.11-1.dsc
linux-5.2.11_5.2.11.orig.tar.gz
linux-headers-5.2.11_5.2.11-1_amd64.deb
linux-image-5.2.11_5.2.11-1_amd64.deb
linux-libc-dev_5.2.11-1_amd64.deb
```

Figure 1.24: Archivos generados en la compilación.

Ahora para poder instalar el Kernel se debe acceder al superusuario con **su -**; debido a que con el - se puede acceder al entorno completo, además de que se reconocerán todas las variables y configuraciones para superusuario, el siguiente comando **dpkg -i linux-image-5.2.11\_5.2.11-1\_amd64.deb**.

```
root@admSist2:/usr/src# dpkg -i linux-image-5.2.11_5.2.11-1_amd64.deb
Selecting previously unselected package linux-image-5.2.11.
(Reading database ... 40035 files and directories currently installed.)
Preparing to unpack linux-image-5.2.11_5.2.11-1_amd64.deb ...
Unpacking linux-image-5.2.11 (5.2.11-1) ...
Setting up linux-image-5.2.11 (5.2.11-1) ...
update-initramfs: Generating /boot/initrd.img-5.2.11
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-5.2.11
Found initrd image: /boot/initrd.img-5.2.11
Found linux image: /boot/vmlinuz-4.19.0-10-amd64
Found initrd image: /boot/initrd.img-4.19.0-10-amd64
Found linux image: /boot/vmlinuz-4.19.0-5-amd64
Found initrd image: /boot/initrd.img-4.19.0-5-amd64
done
root@admSist2:/usr/src# shutdown -r now
packet_write_wait: Connection to UNKNOWN port 65535: Broken pipe
```

Figure 1.25: Instalación del Kernel.

Una vez que finalice la instalación se realiza un **shutdown -r now** para reiniciar el servidor, lo cual cerrara la conexión, por lo que se deba reingresar. Una vez que se esta conectado nuevamente al servidor se realiza un **uname -a** para comprobar que el kernel se instalo correctamente, lo cual en este caso fue asi y se puede ver en la siguiente imagen.

```
-----  
alumno@admSist2:~$ uname -a  
Linux admSist2 5.2.11 #1 SMP Wed Oct 14 15:42:00 -03 2020 x86_64 GNU/Linux  
alumno@admSist2:~$ uname -r  
5.2.11  
alumno@admSist2:~$ █
```

Figure 1.26: Kernel instalado correctamente.

## 2. *Lección 2: Administración de servicios - Acceso Remoto*

### 2.1 Contexto General

### 2.2 Telnet

#### 2.2.1 Trabajo Realizado

Al igual que en la subsección 1.4 se hizo uso del servidor facilitado por el profesor, en donde se requiere trabajar con un compañero, siendo la maquina de Kevin Aguilar con la cual se realizaría la conexión.



Se inicio por el acceso al servidor y una vez en nuestra maquina se instalo Telnet con el siguiente comando **apt-get install telnetd -y**, estando como superusuario, de la misma manera que se muestra en la siguiente imagen.

```
root@admSist2:/home/alumno# apt-get install telnetd -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  linux-image-4.19.0-5-amd64
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
  libevent-2.1-6 openbsd-inetd tcpd update-inetd
The following NEW packages will be installed:
  libevent-2.1-6 openbsd-inetd tcpd telnetd update-inetd
0 upgraded, 5 newly installed, 0 to remove and 5 not upgraded.
Need to get 310 kB of archives.
After this operation, 852 kB of additional disk space will be used.
Get:1 http://ftp.cl.debian.org/debian buster/main amd64 libevent-2.1-6 amd64 2.1.8-stable-4 [177 kB]
Get:2 http://ftp.cl.debian.org/debian buster/main amd64 update-inetd all 4.49 [27.8 kB]
Get:3 http://ftp.cl.debian.org/debian buster/main amd64 tcpd amd64 7.6.q-28 [23.8 kB]
Get:4 http://ftp.cl.debian.org/debian buster/main amd64 openbsd-inetd amd64 0.20160825-4 [36.4 kB]
Get:5 http://ftp.cl.debian.org/debian buster/main amd64 telnetd amd64 0.17-41.2 [44.9 kB]
Fetched 310 kB in 1s (209 kB/s)
```

Figure 2.1: Instalación de Telnet.

Una vez realizada la instalación se comprobó el estado de telnet a través del comando **systemctl status inetd** tal cual se ve en la imagen 2.2.

```
root@admSist2:/home/alumno# systemctl status inetd
● inetd.service - Internet superserver
   Loaded: loaded (/lib/systemd/system/inetd.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-10-20 11:16:29 -03; 1min 16s ago
     Docs: man:inetd(8)
  Main PID: 2609 (inetd)
    Tasks: 1 (limit: 2354)
   Memory: 524.0K
    CGroup: /system.slice/inetd.service
            └─2609 /usr/sbin/inetd

Oct 20 11:16:29 admSist2 systemd[1]: Starting Internet superserver...
Oct 20 11:16:29 admSist2 systemd[1]: Started Internet superserver.
```

Figure 2.2: Estado de Telnet.

Luego se crea un usuario para que un compañero pudiera realizar la actividad y como el con quien se esta trabajando ya la realizo se opta por créale uno a Mauricio Gonzales

debido a que carece de uno. La manera de crear un usuario es a través del comando **useradd NombreUsuario** además de crear un usuario se le asigna una contraseña a este con **passwd NombreUsuario** aquí se pide una confirmar la contraseña para que se cree correctamente. La ejecución de estos comandos se puede ver en la imagen 2.4 y en las imágenes 2.3 y 2.5 se puede ver los usuarios antes y después de realizar la creación respectivamente.

```
root@admSist2:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:104:110::/nonexistent:/usr/sbin/nologin
sshd:x:105:65534::/run/sshd:/usr/sbin/nologin
alumno:x:1000:1000:alumno,,,:/home/alumno:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
telnetd:x:106:112::/nonexistent:/usr/sbin/nologin
```

Figure 2.3: Antes de la creación del usuario.

```
root@admSist2:~# useradd mgonzales
root@admSist2:~# passwd mgonzales
New password:
Retype new password:
passwd: password updated successfully
```

Figure 2.4: Creación del usuario con su contraseña.

```

root@admSist2:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:104:110::/nonexistent:/usr/sbin/nologin
sshd:x:105:65534::/run/sshd:/usr/sbin/nologin
alumno:x:1000:1000:alumno,,,:/home/alumno:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
telnetd:x:106:112::/nonexistent:/usr/sbin/nologin
mgonzales:x:1001:1001::/home/mgonzales:/bin/sh

```

Figure 2.5: Despues de crear el usuario.

Ahora se realiza la conexcion al compañero por lo que se le pide los datos del usuario creado para esto, y se ingresa con el comando **telnet IPdelcompañero** en donde luego se pide un usuario y una contraseña los cuales son los creados por el compañero, si estos se ingresan correctamente se debe ver algo similar a la imagen [2.6](#)

---

```

alumno@admSist2:~$ telnet 10.1.1.31
Trying 10.1.1.31...
Connected to 10.1.1.31.
Escape character is '^]'.
Debian GNU/Linux 10
admSist1 login: acofre
Password:
Linux admSist1 4.19.0-10-amd64 #1 SMP Debian 4.19.132-1 (2020-07-24) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
acofre@admSist1:~$

```

Figure 2.6: Conexión Realizada.

Ahora comprobamos las conexiones activas a traves del comando **who** lo cual debe mostrar 3 conexiones si el compañero esta conectado. Esto se ve en la imagen 2.7.

```
acofre@admSist1:~$ who
alumno pts/0      2020-10-21 17:26 (10.1.1.71)
alumno pts/1      2020-10-21 17:30 (10.1.1.71)
acofre pts/2      2020-10-21 17:45 (stgo-10.1.1.32.in.utralca.cl)
acofre@admSist1:~$
```

Figure 2.7: Conexiones Activas.

## 2.3 SSH

### 2.3.1 Trabajo Realizado

Esta parte se divide en tres actividades, copiar archivos en un servidor remoto mediante SCP, montar localmente el home de un usuario vía ssh y generar una relación de confianza para que al momento de realizar una conexión al servidor remoto.

#### 2.3.1.1 SCP

Aquí se comienza por realizar la conexión al usuario creado por el compañero para obtener la ubicación de principal a través de **pwd** para luego comprobar que esta no tiene contenido gracias al uso de **ls**, tal cual se ve en la imagen 2.8

---

```
acofre@admSist1:~$ pwd
/home/acofre
acofre@admSist1:~$ ls
acofre@admSist1:~$
```

Figure 2.8: Comprobar contenido previo a la transferencia.

Una vez comprobado el contenido de la carpeta, se regresa al servidor remoto "personal" y se transfiere el comprimido que se encuentra ahí, a través del comando **scp archivo usuario remoto@ip:ubicación**, en donde se pedirá ingresar la clave del usuario para poder hacer efectiva la transferencia. La utilización del comando en este caso se ve en la imagen 2.9

```
alumno@admSist2:~$ scp linux-5.2.11.tar.xz acofre@10.1.1.31:/home/acofre
acofre@10.1.1.31's password:
linux-5.2.11.tar.xz
alumno@admSist2:~$
```

100%

Figure 2.9: Transferencia de un archivo al servidor remoto.

Ahora nuevamente se ingresa al usuario remoto para comprobar que la transferencia se realizó con éxito comprobando el contenido con **ls**.

```
acofre@admSist1:~$ ls
linux-5.2.11.tar.xz
acofre@admSist1:~$
```

Figure 2.10: Comprobar la transferencia.

### 2.3.1.2 SSHFS

En esta parte se inicia por la instalación de sshfs a través del comando **apt-get install sshfs** en el dispositivo personal como superusuario, tal cual se ve en la imagen 2.11

```

root@Andres:/home/andres# apt-get install sshfs
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  fop gstreamer0.10-plugins-base icc-profiles-free java-wrappers libaec0 libbatik-java libbcprov-java libcofoja-java lib
  libcommons-math-java libconcurrent-java libfop-java libgoogle-gson-java libgstreamer-plugins-base0.10-0 libgstreamer0.
  libhsjdk-java libjama-java libjargs-java libjcommon-java libjfreechart-java libjgraph-java libjgrapht0.8-java libjhd
  libjide-oss-java libjlibeps-java libngs-java libngs-sdk-dev libngs-sdk1 libnvidia-egl-wayland1 libsaxon-java libsnappy
  libswing-layout-java libsz2 libxmlgraphics-commons-java linux-image-3.16.0-5-amd64 nvidia-egl-wayland-common
Utilice «apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
  sshfs
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 292 no actualizados.
Se necesita descargar 43,1 kB de archivos.
Se utilizarán 116 kB de espacio de disco adicional después de esta operación.
Des:1 http://ftp.cl.debian.org/debian stretch/main amd64 sshfs amd64 2.8-1 [43,1 kB]
Descargados 43,1 kB en 0s (136 kB/s)
Seleccionando el paquete sshfs previamente no seleccionado.
(Leyendo la base de datos ... 286132 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../archives/sshfs_2.8-1_amd64.deb ...
Desempaquetando sshfs (2.8-1) ...
Procesando disparadores para man-db (2.7.6.1-2) ...
Configurando sshfs (2.8-1) ...

```

Figure 2.11: Instalación de SSHFS.

Una vez instalado se accede al servidor remoto y crea un carpeta llamada Anclar con el comando **mkdir Anclar** (Ver Imagen 2.12) luego de esto se regresa a la terminal normal es decir la del dispositivo y crea una carpeta en /mnt de la siguiente manera: **mkdir /mnt/Remoto**.

```

root@admSist2:/home/alumno# mkdir Anclar
root@admSist2:/home/alumno# ls
Anclar  linux-5.2.11.tar.xz
root@admSist2:/home/alumno# cd Anclar
root@admSist2:/home/alumno/Anclar# ls
root@admSist2:/home/alumno/Anclar# █

```

Figure 2.12: Creación de la carpeta en el host remoto.

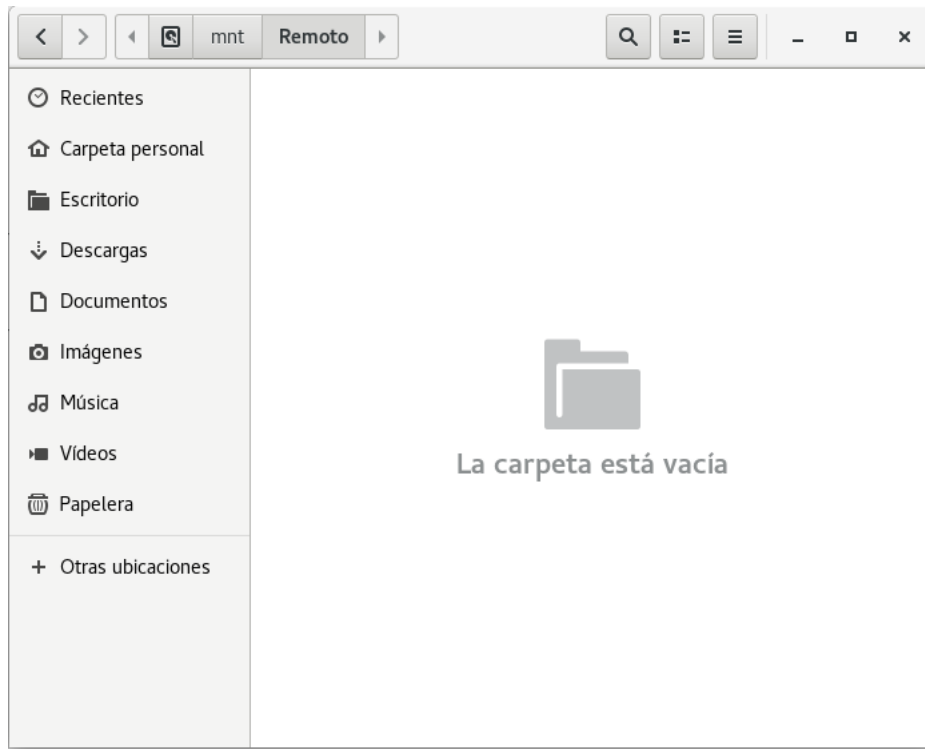


Figure 2.13: Contenido antes de realizar el anclaje.

Luego de esto se ancla a través del comando **sshfs -o allow\_other, default\_permissions hostremoto:/direccionHostRemoto /direccionEnDispositivoPersonal** (Ver Imagen 2.14), este comando en una primera instancia arroja un mensaje que no permite la conexión de otros debido a que el archivo fuse.conf ubicado en /etc/ la línea **user\_allow\_other** esta comentada por lo que se debe quitar el hashtag (#) para que el comando funcione.

```
(base) andres@Andres:~$ sshfs -o allow_other,default_permissions admsist-2:/home/alumno/Anclar /mnt/Remoto/
admsist@bioinfo.utalca.cl's password:
alumno@10.1.1.32's password:
(base) andres@Andres:~$
```

Figure 2.14: Se ancla el home del host remoto.

Ahora en otra terminal se accede a la carpeta creada en el host remoto y se crea un archivo con vi, y al observar en el directorio del "computador normal" se puede ver que el

archivo aparece en la carpeta (Ver Imagen 2.15), el cual puede ser editado y los cambios se verán en el archivo del host remoto.

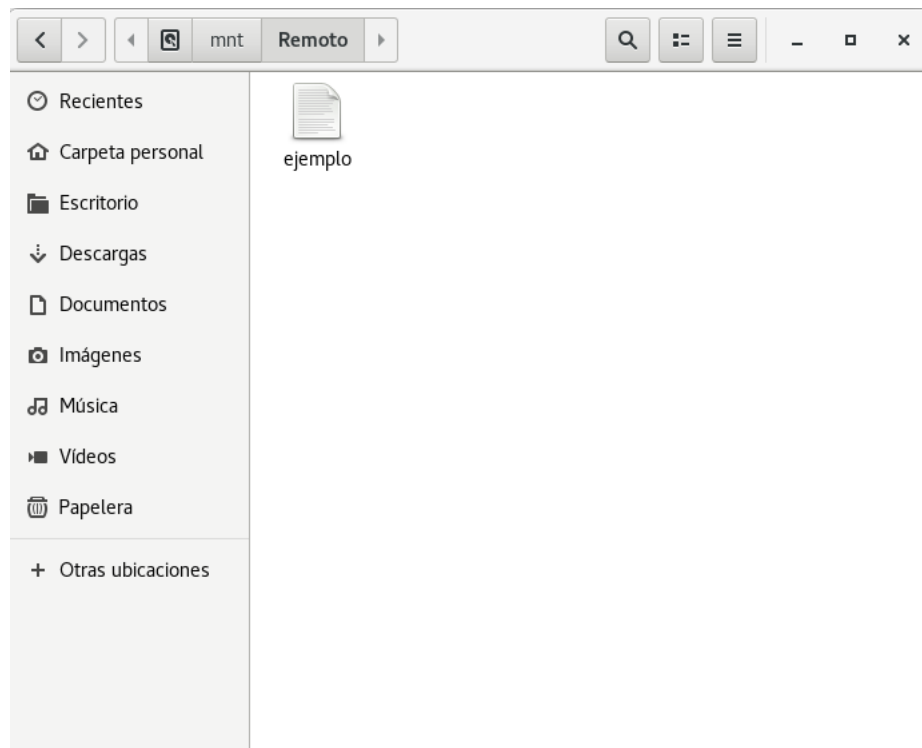


Figure 2.15: Después de anclar.

Para poder desmontar se utiliza **umount /ubicaciónEnElComputador**.

### 2.3.1.3 Relación de Confianza

Para poder crear la relación de confianza y que no se pida clave al momento de conectarse al host remoto primero se debe generar una key a través del comando **ssh-keygen -t rsa** (Ver Imagen 2.16)



```
(base) andres@Andres:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/andres/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/andres/.ssh/id_rsa.
Your public key has been saved in /home/andres/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:Lk9AmBrKAmk01oS2dkFd1PGw78YjP4ljWEPT1tWFfGY andres@Andres
The key's randomart image is:
+---[RSA 2048]---+
|  oo. oo.o. . .+ |
| +o. + .+ o.E |
| =oo.+ . .... .+ |
| B+ + . o.o . |
| +oo . S o. |
| . o oo |
| . =.o=. |
| = ++o. |
| o ... |
+----[SHA256]-----+
```

Figure 2.16: Generación de la Key.

Ahora se crea una carpeta para almacenar a key del host remoto a través del comando **ssh hostRemoto mkdir -p .ssh** (Ver Imagen 2.17) y luego de ingresar las contraseñas se le asigna una de las contraseñas autorizadas con **cat .ssh/id\_rsa.pub — ssh hostRemoto 'cat && .ssh/authorized\_keys'**. Con esto se puede saltar la ultima contraseña (Ver Imagen 2.18)

```
+----[SHA256]-----+
(base) andres@Andres:~$ ssh admsist-2 mkdir -p .ssh
admsist@bioinfo.utalca.cl's password:
alumno@10.1.1.32's password:
(base) andres@Andres:~$ cat .ssh/id_rsa.pub | ssh admsist-2 'cat >> .ssh/authorized_keys'
admsist@bioinfo.utalca.cl's password:
alumno@10.1.1.32's password:
(base) andres@Andres:~$
```

Figure 2.17: Almacenaje de las key y asignación de las claves.

```
(base) andres@Andres:~$ ssh admsist-2
admsist@bioinfo.utalca.cl's password:
Linux admSist2 5.2.11 #1 SMP Wed Oct 14 15:42:00 -03 2020 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Nov  5 12:01:42 2020 from 10.1.1.71
alumno@admSist2:~$
```

Figure 2.18: Se salta la ultima contraseña.

Ahora se debe repetir el paso anterior pero con respecto al host remoto de la primera conexión, en este caso el host remoto seria `admsist@bioinfo.utalca.cl` y el comando utilizado es `ssh admsist@bioinfo.utalca.cl mkdir -p .ssh` y luego `cat .ssh/id_rsa.pub — ssh admsist@bioinfo.utalca.cl 'cat && .ssh/authorized_keys'` (Ver Imagen 2.19)

```
(base) andres@Andres:~$ ssh admsist@bioinfo.utalca.cl mkdir -p .ssh
admsist@bioinfo.utalca.cl's password:
(base) andres@Andres:~$ cat .ssh/id_rsa.pub | ssh admsist@bioinfo.utalca.cl 'cat >> .ssh/authorized_keys'
admsist@bioinfo.utalca.cl's password:
(base) andres@Andres:~$
```

Figure 2.19: Almacenaje de las key y asignación de las claves para la otra conexión

Ahora si se conecta a traves de ssh al host remoto no se pedira ninguna contraseña y esto se ve en la imagen 2.20

```
(base) andres@Andres:~$ ssh admsist-2
Linux admSist2 5.2.11 #1 SMP Wed Oct 14 15:42:00 -03 2020 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Nov  5 12:51:02 2020 from 10.1.1.71
alumno@admSist2:~$
```

Figure 2.20: Relación de Confianza creada

### 3. *Lección 3: Administración de servicios- Transferencia de Datos*

#### 3.1 Contexto General

#### 3.2 Práctica: TFTP

#### 3.3 Práctica: FTP

## 4. *Problemas Encontrados*

### 4.1 **Lección 1**

#### 4.1.1 **Practica: Compilación de GCC**

Para esta parte el problema que se presento fue con la dirección en la que se trabajaba dado que, al momento de ejecutar la configuración de la primera librería, esta no reconoce el espacio del nombre en la carpeta principal. Por lo que se renombra las carpetas que contengan un espacio para que esto no pase nuevamente.

#### 4.1.2 **Practica: Kdtree**

Para esta parte no se encontraron complicaciones.

#### 4.1.3 **Practica: Kernel**

Aquí el problema encontrado fue en relación a la conexión con el servidor, debido a que la compañía de internet estaba bloqueando algunos puertos por "temas de seguridad" cuando

se les pregunto la razón. Para poder solucionar esto y no tener que acceder con los datos del celular ya que el proceso seria mucho mas lento debido a la velocidad, se comunico con un encargado para poder habilitar el servicio, entregando la IP publica del destinatario, el protocolo que se utiliza y el puerto en uso.

## **4.2 Lección 2**

### **4.2.1 Practica: Telnet**

En esta parte no se encontraron problemas.

### **4.2.2 Practica: SSH**

Problemas con el entendimiento sobre como realizar el trabajo, los cuales se solucionaron al trabajar con el compañero.

## **4.3 Lección 3**

### **4.3.1 Practica: TFTP**

### **4.3.2 Practica: FTP**