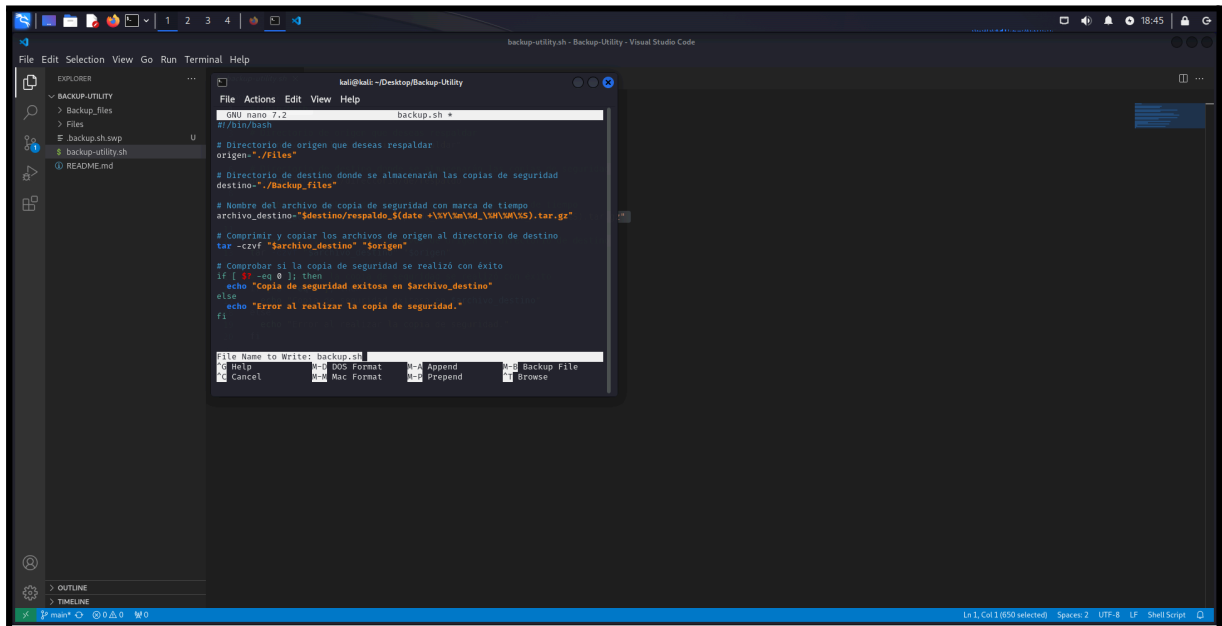


## Gabriel Ramos – gabe-cloud CS 2 Backup-Utility

Primeramente se hizo un duplicado del script backup-utility.sh mediante nano backup.sh para luego ser modificado según los parámetros de origen y destino que iban a ser usados, donde primeramente se agregaron directorios locales para primeramente hacer una prueba de que el script funcionaba correctamente.

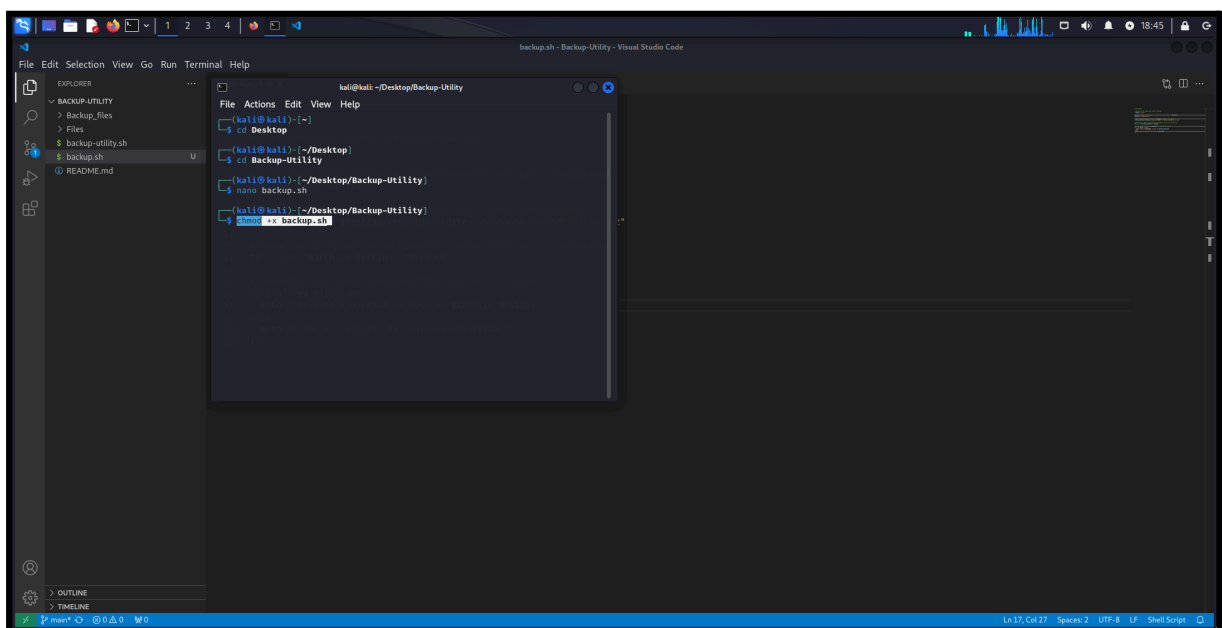


The screenshot shows the Visual Studio Code editor with a file named backup.sh open. The script content is as follows:

```
#!/bin/dash
# Directorio de origen que deseas respaldar
origen=".."
# Directorio de destino donde se almacenarán las copias de seguridad
destino="Backup_files"
# Nombre del archivo de copia de seguridad con marca de tiempo
archivo_destino="$destino/respaldo_$(date +%Y/%m/%d_%H/%M/%S).tar.gz"
# Comprimir y copiar los archivos de origen al directorio de destino
tar -czvf "$archivo_destino" "$origen"
# Comprobar si la copia de seguridad se realizó con éxito
if [ $? -eq 0 ]; then
    echo "Copia de seguridad exitosa en $archivo_destino"
else
    echo "Error al realizar la copia de seguridad."
fi
```

A context menu is visible over the script, with options: Help, DOS Format, Append, Backup File, Cancel, Mac Format, Prepend, and Browse.

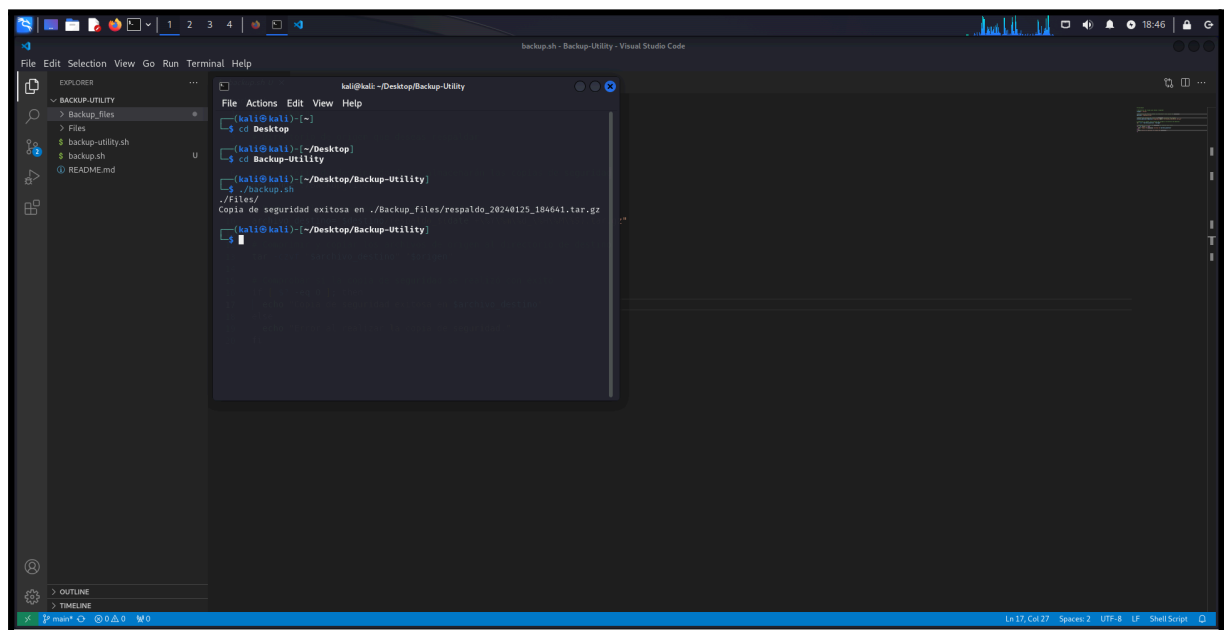
Luego se dieron permisos de ejecución al script mediante el comando `chmod +x backup.sh`



The screenshot shows the Visual Studio Code editor with a terminal window open. The terminal displays the following commands and output:

```
(kali@kali)~$ cd Desktop
(kali@kali)~/Desktop$ cd Backup-Utility
(kali@kali)~/Desktop/Backup-Utility$ nano backup.sh
(kali@kali)~/Desktop/Backup-Utility$ chmod +x backup.sh
```

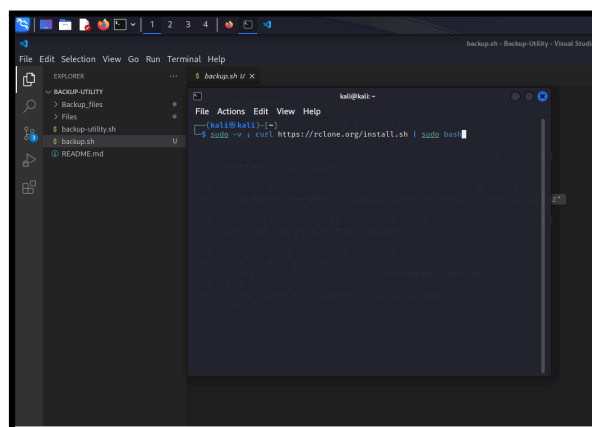
Una vez configurado todo, se hizo la prueba local del script de backup.



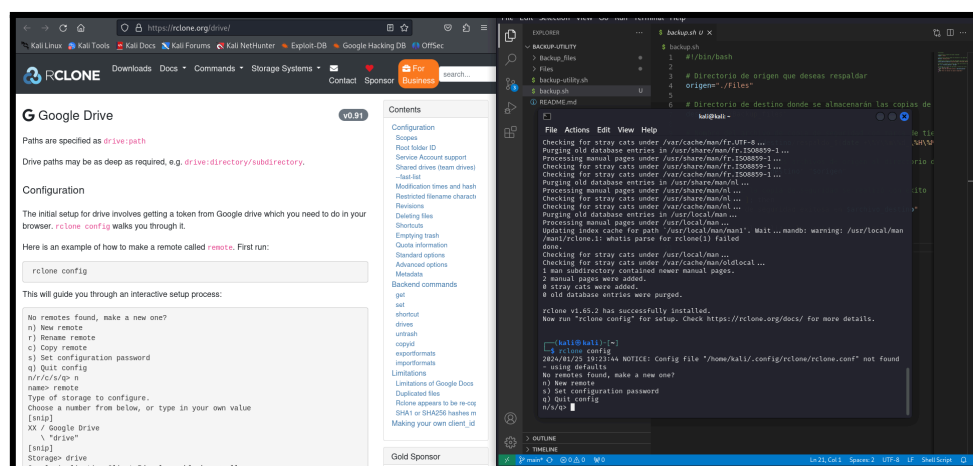
```
kali@kali: ~/Desktop/Backup-Utility
$ cd Desktop
$ cd Backup-Utility
$ ./backup.sh
Copia de seguridad exitosa en ./Backup_files/respaldo_20240125_184641.tar.gz
$
```

**Funcionó correctamente!**

Seguidamente como se debe subir a la nube se hizo uso de “rclone” el cual fue instalado y configurado el equipo para hacer el backup a Google Drive.

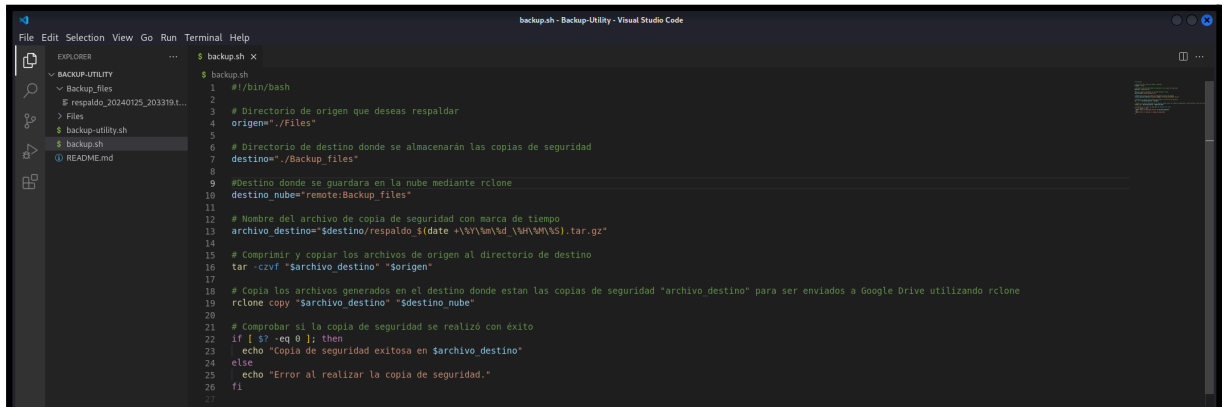


```
kali@kali: ~
$ sudo -v
$ curl https://rclone.org/install.sh | sudo bash
```

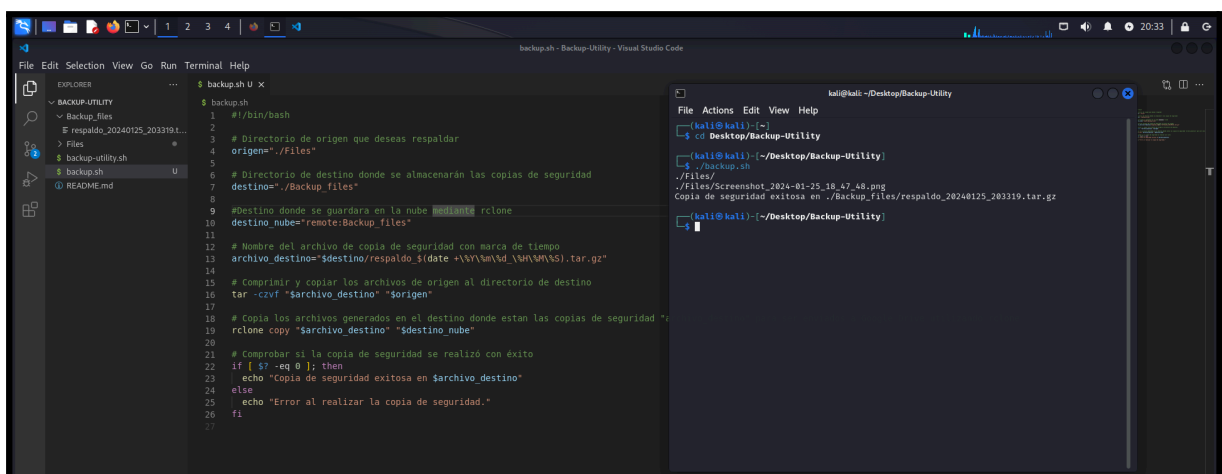


The screenshot shows the rclone website's configuration page on the left and a terminal window on the right. The website page is titled "Google Drive" and provides instructions on how to configure rclone for Google Drive. It includes a "Configuration" section with a list of steps: "n) New remote", "r) Remote name", "s) Set configuration password", "q) Quit config", and "n) New remote". The terminal window shows the user running the "rclone config" command, which prompts them to enter a remote name, a remote type, and a configuration password.

Una vez fue configurado completamente se procedió a modificar el código con unos cambios para que funcionara con rclone, donde esté primeramente hace el backup localmente y una vez es hecho el backup lo envía a Google Drive mediante la conexión antes configurada con rclone.

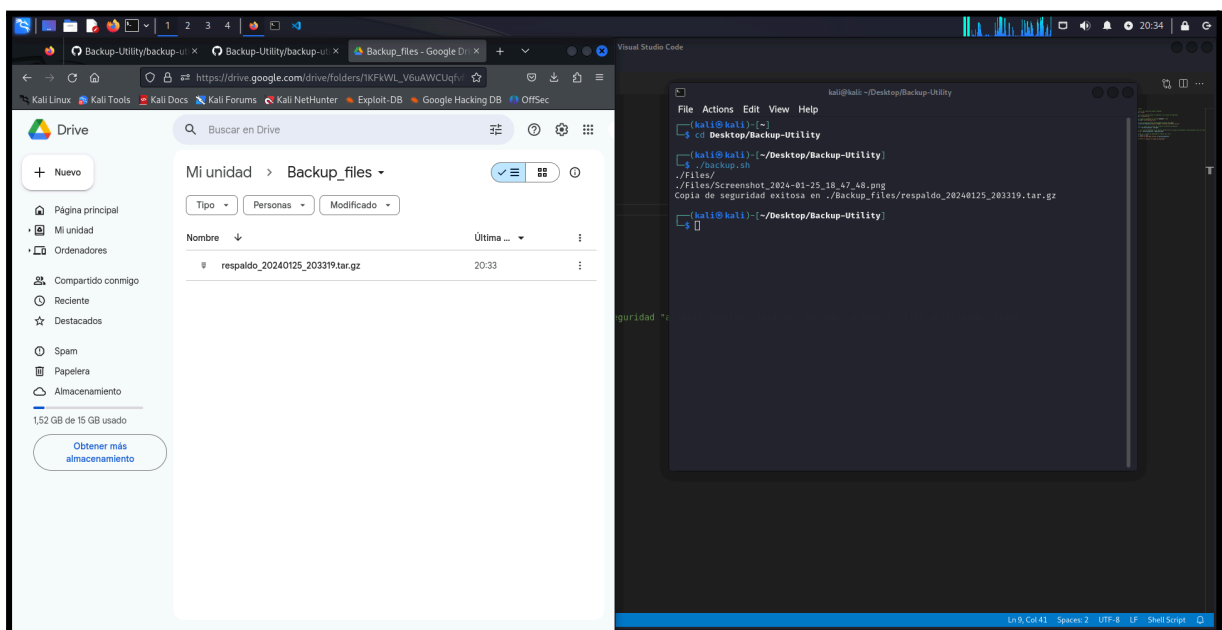


```
1 #!/bin/bash
2
3 # Directorio de origen que deseas respaldar
4 origen="/.Files"
5
6 # Directorio de destino donde se almacenarán las copias de seguridad
7 destino="/Backup_files"
8
9 #Destino donde se guardara en la nube mediante rclone
10 destino_nube="remote:Backup_files"
11
12 # Nombre del archivo de copia de seguridad con marca de tiempo
13 archivo_destino="$destino/respaldo_$(date +%Y%m%d_%H%M%S).tar.gz"
14
15 # Comprimir y copiar los archivos de origen al directorio de destino
16 tar -czvf "$archivo_destino" "$origen"
17
18 # Copia los archivos generados en el destino donde estan las copias de seguridad "archivo_destino" para ser enviados a Google Drive utilizando rclone
19 rclone copy "$archivo_destino" "$destino_nube"
20
21 # Comprobar si la copia de seguridad se realizó con éxito
22 if [ $? -eq 0 ]; then
23     echo "Copia de seguridad exitosa en $archivo_destino"
24 else
25     echo "Error al realizar la copia de seguridad."
26 fi
```



```
1 #!/bin/bash
2
3 # Directorio de origen que deseas respaldar
4 origen="/.Files"
5
6 # Directorio de destino donde se almacenarán las copias de seguridad
7 destino="/Backup_files"
8
9 #Destino donde se guardara en la nube mediante rclone
10 destino_nube="remote:Backup_files"
11
12 # Nombre del archivo de copia de seguridad con marca de tiempo
13 archivo_destino="$destino/respaldo_$(date +%Y%m%d_%H%M%S).tar.gz"
14
15 # Comprimir y copiar los archivos de origen al directorio de destino
16 tar -czvf "$archivo_destino" "$origen"
17
18 # Copia los archivos generados en el destino donde estan las copias de seguridad "archivo_destino" para ser enviados a Google Drive utilizando rclone
19 rclone copy "$archivo_destino" "$destino_nube"
20
21 # Comprobar si la copia de seguridad se realizó con éxito
22 if [ $? -eq 0 ]; then
23     echo "Copia de seguridad exitosa en $archivo_destino"
24 else
25     echo "Error al realizar la copia de seguridad."
26 fi
```

```
kali@kali:~/Desktop/Backup-Utility
$ ./backup.sh
Copia de seguridad exitosa en ./Backup_files/respaldo_20240125_203319.tar.gz
```



Google Drive interface showing the file 'respaldo\_20240125\_203319.tar.gz' in the 'Backup\_files' folder.

```
File Actions Edit View Help
[kali@kali:~/Desktop/Backup-Utility]
$ ./backup.sh
Copia de seguridad exitosa en ./Backup_files/respaldo_20240125_203319.tar.gz
```