

Procedimiento de copias de seguridad.

Adrian Navarro SR2A

Índice

I-Introducción

- 2-Contenidos
 - a. Configuración de Rclone
 - b. Control básico de rclone
 - c. Funcionalidad del procedimiento
 - d. Crontab
 - e. Recuperación

I- Introducción

En este documento se detalla la configuración de la herramienta "Rclone" utilizada para la gestión de archivos en sistemas de almacenamiento en la nube.

Clarificación de los scripts de copias de las copias de seguridad, así como su puesta en marcha y automatización con "crontab"

También se incluye un ejemplo sencillo de recuperación de los datos.

2- Contenidos

a. Configuración de Rclone

Rclone es una herramienta de gestión de archivos en cuentas de almacenamiento en la nube. Es de uso libre y está disponible en las principales distribuciones de Linux.

Para instalarla en un sistema Centos:

sudo yum install rclone

Para instalarla en un sistema Ubuntu:

sudo apt install rclone

Una vez instalado Rclone metemos el comando *sudo rclone config* (los perfiles de rclone se guardan en los home de cada usuario, por lo que si queremos que nuestro usuario tenga también acceso a la cuenta de almacenamiento tendremos que repetir el proceso de configuración, pero sin el "sudo" *rclone config*).

```
[inf00047@estudios ~]$ sudo rclone config
No remotes found - make a new one
n) New remote
s) Set configuration password
q) Quit config
n/s/q>
```

Como se puedo ver no tenemos ningún "remote" guardado aun por lo que pondremos "n" new remote y le daremos el nombre que deseemos. En mi caso he escogido cloud ya que así se hace referencia en los scripts.

```
[inf00047@estudios ~]$ sudo rclone config
No remotes found - make a new one
n) New remote
s) Set configuration password
q) Quit config
n/s/q> n
name> cloud
```

Se nos desplegara la lista de los servicios en la nube a los que tiene acceso rclone. Buscamos el equivalente al que nosotros deseamos introducimos el número que le hace referencia y continuamos.

Se yo he escogido OneDrive de Microsoft, los procesos de configuración pueden variar según la plataforma.

```
Type of storage to configure.

Enter a string value. Press Enter for the default ("").

Choose a number from below, or type in your own value

1 / A stackable unification remote, which can appear to merge the contents of several remotes
\"union"

2 / Alias for a existing remote
\"alias"
          "alias"
Amazon Drive
"amazon cloud drive"
Amazon Cloud drive"
Amazon S3 Compliant Storage Provider (AWS, Alibaba, Ceph, Digital Ocean, Dreamhost, IBM COS, Minio, etc)
          Backblaze B2
"b2"
  6 / Box
\ "box"
          Cache a remote
"cache"
  8 / Dropbox
\ "dropbox"
           Encrypt/Decrypt a remote
         / Encrypt/Decrypt a remote
\( \text{"crypt"}
\)
/ "FTP Connection
\"ftp"
/ Google Cloud Storage (this is not Google Drive)
\"google cloud storage"
/ Google Drive
\"drive"
/ Hubic
\"hubic
\"hubic"
/ Jottacloud
\"jottacloud
\"jottacloud"
/ Koofr
\"koofr"
/ Local Disk
12 /
13 /
           Local Disk
"local"
          "Total
Mega
"mega"
Microsoft Azure Blob Storage
"azureblob"
           Microsoft OneDrive
"onedrive"
20 / OpenDrive
\ "Opendrive"
21 / Openstack Swift (Rackspace Cloud Files, Memset Memstore, OVH)
\ "swift"
22 / Pcloud
22 / Pcloud
\ "pcloud"
23 / QingCloud Object Storage
\ "qingstor"
24 / SSH/SFTP Connection
\ "sftp"
25 / Webdav
\ "webdav"
            Yandex Disk
           "yandex"
http Connection
    torage> _
```

A continuación, nos aparecerá dos apartados en los cuales podemos darle a entre y tomar los valores por defecto.

Sin embargo, también nos saldrá un mensaje de si queremos editar la configuración avanzada. Los valores por defecto son los que se han utilizado y no hay problemas por lo que le indicamos "n" no.

```
Storage> 19

** See help for onedrive backend at: https://rclone.org/onedrive/ **

Microsoft App Client Id

Leave blank normally.

Enter a string value. Press Enter for the default ("").

client_id>

Microsoft App Client Secret

Leave blank normally.

Enter a string value. Press Enter for the default ("").

client_secret>

Edit advanced config? (y/n)

y) Yes

n) No

y/n>
```

El siguiente mensaje nos pregunta se estamos trabajando con una maquina con interfaz gráfica o si estamos trabajando con una maquina sin interfaz gráfica (o por ssh)

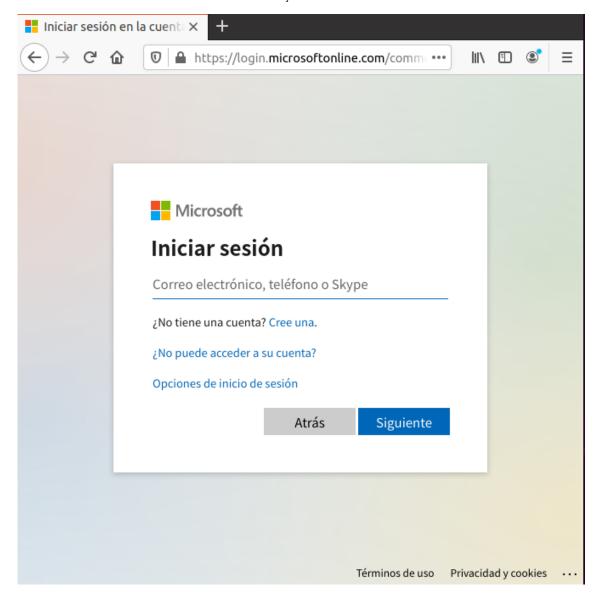
Si tenemos interfaz ponemos "y" y si no tenemos ponemos "n". En mi caso no tenía. (El proceso es el mismo pero con un pasito intermedio)

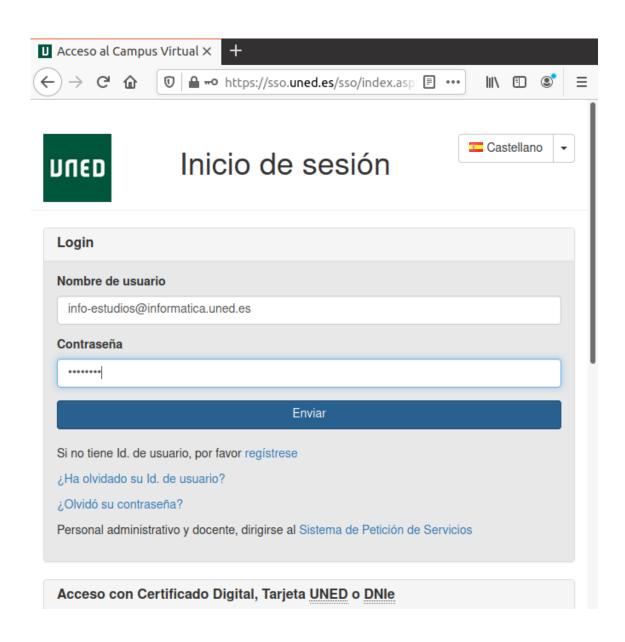
```
Edit advanced config? (y/n)
y) Yes
n) No
y/n> n
Remote config
Use auto config?
* Say Y if not sure
* Say N if you are working on a remote or headless machine
y) Yes
n) No
y/n> _
```

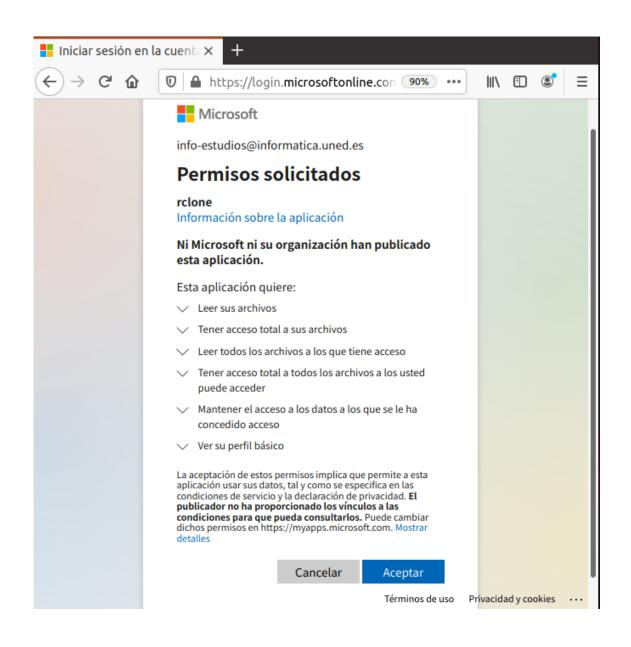
En otra máquina Linux en la que tengamos acceso a la interfaz grafica, instalamos rclone y metemos el comando rclone authorize "onedrive"

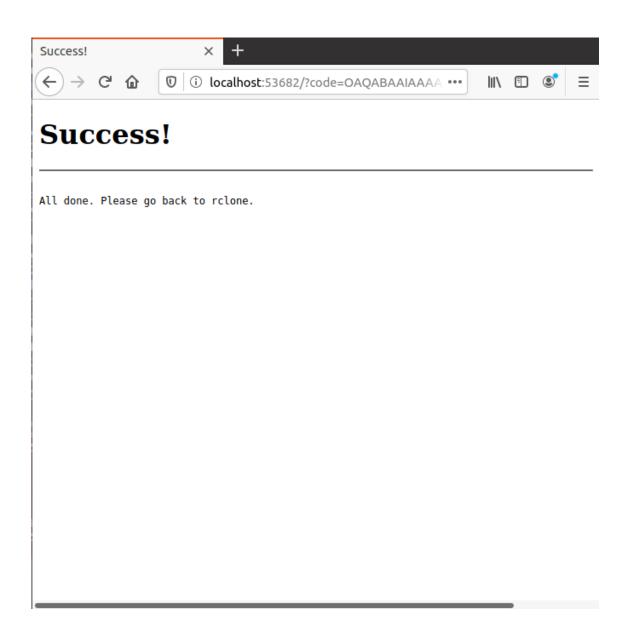
```
odingodin-VirtualBox:~$ rclone authorize "onedrive"
If your browser doesn't open automatically go to the following link: http://127.0.0.1:53682/auth?
state=3Tsu1B_Dq5_7peB7bWlxHA
Log in and authorize rclone for access
Waiting for code...
```

Esto debiera abrirnos el navegador en la con la dirección del portal de inicio de sesión de Microsoft. Nos validamos y volvemos a la terminal.









Waiting for code.. Paste the following into your remote machine ---> {"access_token":"eyJ0eXAi0iJKV1QiLCJub25jZSI6IlNsTUUzRURWcjZsTHdkUG9SUF9wR2lKUU1neUoxQVlkUkR4S0dT czk4VE0iLCJhbGci0iJSUzI1NiIsIng1dCI6IkN0VHVoTUptRDVNN0RMZHpEMnYyeDNRS1NSWSISImtpZCI6IkN0VHVoTUptR DVNNORMZHPEMnYyeDNRS1NSWSJ9.eyJhdWQi0ilwMDAwMDAwMY0wMDAwLTAwMDAtYzAwMCOwMDAwMDAwMDAwMDAiLCJpc3Mi0 iJodHRwczovL3N0cy53aW5kb3dzLm5ldC9iNTU4NzA0OC1hMDcwLTQzYWQtYjY1NC0yMGEzMWU5YjJkOTkvIiwiaWF0IjoxNT kwnzQINTk0LCJuYmYiOjE10TA3NDU10TQSImV4cCI6MTU5MDc0OTQ5NCwiYWNjdCI6MCwiYWNyIjoiMSISImFpbyI6IjQyZGd ZTERhek5QSUkyRHltRmYyNFI4ajlkQnpjK3YzYU9X0FBucEVXK0QxaFZXc1FxY0IiLCJhbXIiolsicHdkIl0sImFwcF9kaXNw bGF5bmFtZSI6InJjbG9uZSIsImFwcGlkIjoiYjE1NjY1ZDktZWRhNi00MDkyLTg1MzktMGVlYzM3NmFmZDU5IiwiYXBwaWRhY 3IiOiIXIiwiZmFtaWx5X25hbWUiOiJDQVJSQVNDTyBNT1NRVUVSQSIsImdpdmVuX25hbWUiOiJDRVNBUIISImlwYWRkci16Ij gwLjMwLjMuNDEiLCJuYW1lIjoiQ0VTQѶIgQ0FSUKFTQ08gTU9TUVVFUKEiLCJVaWQi0iI4ZTc5MZA0ZC0xMZBKLTQ4ZmYtOWF lOS01YzJkZjY1ZGQ0Y2UiLCJwbGF0ZiI6IjE0IiwicHVpZCI6IjEwMDMyMDAwQzE40UYyNzYiLCJzY3Ai0iJGaWxlcy5SZWFk IEZpbGVzLlJlYWQuQWxsIEZpbGVzLlJlYWRXcml0ZSBGaWxlcy5SZWFkV3JpdGUuQWxsIHByb2ZpbGUgb3BlbmlkIGVtYWlsI iwic3ViIjoiSm40Vm53dTNfYV9VX3N1ZnpIZWY1bzE3aVd0WE1YN1R3aHRRZTc5Q3FCYyIsInRlbmFudF9yZWdpb25fc2Nvc0 UiOiJFVSIsInRpZCI6ImI1NTg3MDQ4LWEwNzAtNDNhZC1iNjU0LTIwYTMxZTliMmQ5OSIsInVuaXF1ZV9uYW1lIjoiaW5mby1 lc3R1ZGlvc0BpbmZvcm1hdGljYS51bmVkLmVzIiwidXBuIjoiaW5mby1lc3R1ZGlvc0BpbmZvcm1hdGljYS51bmVkLmVzIiwi dXRpIjoicWFPbk9JbuVTVXUzUDdScFllVXFBQSIsInZlci16IjEuMCIsInhtc19zdC16eyJzdWIi0iJtMVdYWlVpSWZZNVRYI kNVZHZWbjVxUW40UmM2ZnRMajYzV3l2ZTNZVĒpJIn0sInhtc190Y2R0IjoxMzY5MjU2NTIyfQ.ICgeiuudooLONdiYrcZwkZJ nnpHkC983HRz2P-DiagTeg9rY9IsiHqg8D_xJFErXn6n_Dbzg4OTmMI7AZh4uTiYbGJTr9htx064Y_MR31ve6VE8PvSfN1vJ mgtozzyCVp8QbyjHRRQ7ehRxc9QPypD8ChB6s8ZXLa3eMw3c65gj3IcIjgcJha3kZ1HxIKAJZrN122nhaVN50QI_s4k-XOwf 0XqSUjYTcQCSmEHOntYKJh3YSmZDPHMDUvjrCx0Nd31L4Ei8tdQTUE1mvP5DS6a5Fzaw93DnN7CoMdriymXcGt3h0nNolTA5D BbFBDaeSb-NlaD9pRT4JqdA6CMt-50RR0em5aBWdJAM11xkn6fVkJLJlI37RGY8oTWHwJgqIjMe6-UH8No9242pHKX778vH8z YiDZfBNF0lZVc4Yfhom2TY1KnEAQ5f0yhUq0v2lTxHDdEY45zERR25I54GKihy3StjNeBCwGVsPqPjUDhRwmGIY806Ed8Sa11 mzJCb7SklHSmFHn4_03LMdd4fkxlFH3xPyHXWno__ipnjz07QJ-NDPtbElGKcPL-1TtWG_X4XoWFKWqqb2auIL-fm4kUCIyjp rWnnXX19rs6qMO2w6l04n3HdP8k7qK-NiL4pzpuMubiuOYEvgNYN5glt4i8cDNLWaYSMSV95lbNQiA8bTBqVLiKzzWY2yZ11Y gMdT8ZqcSj8DsO9tOD2Lb-7yJyG8rbmQWCyb5MClCU6M0NNAlhWsVZcKXAyx1I40FGT_W0jG7BQGVn5swVtjxc10-aN153Yuo k2DAIuY0-8oYkvKmd4MLL--sk6Sba8gQC7IToYOdUX2-h5aZfdQizcMkPRW0Pj47wc6TKm9tsvP10Cm-L6IVtpb2xJChQYxJxuSZcCzzad4XwK9q0Y6jUAK29bZjn2xbEW6PFhXKB1iSI6o6KrLF5Rz6qrfuvUrEHloM1MS4ICjawI2vmj6BWiXli9YOnMDjS fy_kdANURbZyXKsHSAA","expiry":"2020-05-29T12:51:34.908040901+02:00"}

Hacemos como se nos pide y copiamos todo el texto que está entre las llaves y lo pegamos en la terminal de la maquina en la que estamos configurando el rclone

Continuamos u nos preguntara que tipo de cuanta de almacenamiento es. Diremos que es Onedrive personal o business "I"

Ahora seleccionamos el disco raíz al que equivaldrá el remoto, en mi caso era la primera opción "0"

```
Found 2 drives, please select the one you want to use:
0: Fotografías del usuario (documentLibrary) id=b!v8HP6C-eIUqD6A8c2GZGKWJbNyXYh7JCjLxY1NTF240_rnHr57yXQ4xERa0MEdTb
1: Logotipos de la organización (documentLibrary) id=b!v8HP6C-eIUqD6A8c2GZGKWJbNyXYh7JCjLxY1NTF241vhsXmnomhQrR2dNpr1c
-1
Chose drive to use:> 0_

Found drive 'root' of type 'documentLibrary', URL: https://unedo365-my.sharepoint.com/User%20Photos
Is that okay?
y) Yes
n) No
y/n> y
```

Confirmamos que todo este correcto y le damos a "y"

Y ya tenemos nuestro remoto configurado

b. Control básico de rclone

Relone tiene sus propios comandos para funcionar, pero son bástate sencillos. Las rutas de los remotos son nombreDeRemoto:directorio I / directorio 2 /

Si queremos copiar algo de nuestra maquina a la nube

Ejemplo:

```
rclone copy /home/odin/test.txt cloud:home/odin/
```

Y si queremos copiar algo de la nube a nuestra maquina el proceso es el mismo, solo hay que invertir las rutas.

Supongamos que queremos eliminar algún elemento de la nube:

```
rclone delete <ruta_de_la_nube>
```

Se queremos ver el contenido de un directorio en la nube

Luego hay un comando un poco contra intuitivo

Este comando te sincroniza el contenido de la rutaI en la ruta2, pero si en la ruta 2 hay algún dato, también lo elimina. De tal manera que realmente te crea un espejo de una ruta en la otra.

c. Funcionalidad del procedimiento

El paquete de scripts cuenta con dos vertientes, la semanal y la diaria. A nivel de estructura son los dos iguales con la única salvedad de que la semanal cuenta con un bloque de código que se asegura de que haya x número de archivos y que su nombre incluye la fecha en la que se realizó la copia. La estructura y filosofía de trabajo es la siguiente:

```
etc/scriptsBK/
  diario
     - mainmk1.sh
      modulos
           apache2
               apache2.sh
               home.sh
           mysql
               mysql.sh
           postgresql
               postgresql.sh
  semanal
      mainmk1.sh
      modulos
           apache2
               apache2.sh
               home.sh
               mysql.sh
```

Como podemos ver se cuenta con un archivo "main.sh" que es el que se ejecutará y luego en el directorio "modulos" se incluirán los distintos servicios de los que se quieren hacer las copias de seguridad.

En "main.sh" contamos con la inclusión previamente comentada de los distintos módulos. Por otro lado, se establecen las variables de la fecha en la que se ejecuta el procedimiento, creación de la carpeta "/backups/", la creación del archivo de logs y la subida a la nube del propio archivo de logs.

El procedimiento está diseñado de tal manera que si se quiere introducir un nuevo servicio sea tan fácil como clonar uno de los módulos ya existentes y modificar un par de variables.

Cada módulo te creara uno o dos directorios "conf" y/o "data", dentro de ellos encontraremos paquetes comprimidos. Dependiendo, encontraremos archivos de datos o de configuración de los servicios. Estos se habrán cogido de un directorio temporal ubicado en el directorio "/tmp/" que a su vez habrán sido copiados ahí automáticamente por el script. De tal manera que si se necesita añadir algún otro archivo de simplemente con añadir un *cp -rp <archivoA_Añadir>* en el script estaría listo.

En algunos casos en los directorios de "data" no bastara con copiar y pegar, ya que por ejemplo con las bases de datos hemos tenido que volcarlas utilizando sus respectivos dump.

Un dato importante a tener en cuenta es que por defecto se comprimirán los paquetes con una extensión .gz.

Continuando con el procedimiento encontramos una estructura de control que nos evaluara el tamaño del paquete y si es mayor de 10GB no los dividirá en partes más pequeños. Esto es por una limitación de la nube de Microsoft que no nos permite subir archivos de más de 10GB.

En los módulos de la vertiente semanal nos encontraremos un último bloque de código que se comprueba el número de archivos en el directorio y si es mayor a 3, eliminara el más antiguo dejando solamente 2 archivos.

Por supuesto el procedimiento te genera una copia de seguridad en el directorio /backups/ y luego también te genera la misma estructura en la nube.

```
backups/
— diario
                     apacheConf_.tar.gz
                    – apacheData .tar.gz
                data
                     homeData_.tar.gz
          log.log
                conf
└─ bk_mysql_conf.tar.gz
                    mysqldump_diario.tar.gz
                   postgresqlConf.tar.gz
                     postgresDump_diario.tar.gz
    semanal
                   — apacheConf_17_06_2020_13_40.tar.gz
— apacheConf_21_06_2020_02_00.tar.gz
                    - apacheData_17_06_2020_13_40.tar.gz
- apacheData_21_06_2020_02_00.tar.gz
                   - homeData_17_06_2020_13_40.tar.gz
                     homeData_21_06_2020_02_00.tar.gz
         log.log
          mysql
                     bk_mysqlConf17_06_2020_13_40.tar.gz
bk_mysqlConf21_06_2020_02_00.tar.gz
                     mysqldump_semanal17_06_2020_13_40.tar.gz
mysqldump_semanal21_06_2020_02_00.tar.gz
          postgresql
                    .
· postgresqlConf17_06_2020_13_40.tar.gz
· postgresqlConf21_06_2020_02_00.tar.gz
                     postgresDump_semanal17_06_2020_13_40.tar.gz
postgresDump_semanal21_06_2020_02_00.tar.gz
```

d. Crontab

Crontab es una herramienta que tiene Linux que nos permite automatizar tareas. Nosotros establecemos cuando han de ejecutarse las tareas. Para editarlo tecleamos *sudo nano /etc/crontab*

```
GNU nano 2.3.1 Fichero: /etc/crontab Modificado

# .------ minute (0 - 59)

# | .------ day of month (1 - 31)

# | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat

# | | | | | |

# * * * * * user-name command to be executed

# * * * * * user-name command to be executed

# * * * * * or oot sh /etc/scriptsBK/diario/mainmk1.sh

2 * * 0 root sh /etc/scriptsBK/semanal/mainmk1.sh

AG Ver ayuda

AG Ver ayuda

AG User Fich

AY Pág Ant

AK CortarTxt

AC Pos actual

AN Salir

AJ Justificar

AN Buscar

AV Pág Sig

AU PegarTxt

AT Ortografía
```

Si nos fijamos, nos dice que el primer campo hace alusión a los minutos, el segundo campo a la hora, el tercero al día del mes, el cuarto al mes y el quinto al día de la semana. Luego está el usuario con el que se ejecutara la tarea y la propia tarea que queremos que realice.

En la imagen vemos que tengo configurado que me ejecute el main.sh diario todos los días de la semana de cada mes a las 02:00 con el usuario root.

```
02 * * * root sh /etc/scriptsBK/diario/mainmk1.sh
```

También tengo configurado que me ejecute el main.sh semanal solo el domingo de todas las semanas de todos los meses a las 02:00 como root.

02 * * * root sh /etc/scriptsBK/diario/mainmk1.sh

e. Recuperación

Dependiendo del servicio, el proceso de recuperación puede variar ligeramente.

Lo primero es ser conscientes de en qué punto nos encontramos. El peor de los casos sería un fallo catastrófico de la máquina que requiriese instalar desde cero.

En este caso lo primero que tendríamos que hacer sería instalar y configurar de nuevo rclone para tener acceso a nuestras copias de seguridad alojadas en la nube y descargarlas. Si a la hora de descargar vemos que lo que se nos descarga es una carpeta, esto se debe a que el paquete pesaría más de 10 GB y se tuvo que dividir.

Para solventar este problema tendremos que juntar las partes. No metemos en la carpeta y ejecutamos

cat directorio_con_las_partes/parte_* > directorio_con_las_partes/data.tar.gz

```
inf00047@odin-VirtualBox:~/cachos/partes$ ls
data.tar.parte_aa data.tar.parte_af data.tar.parte_ak data.tar.parte_ap
data.tar.parte_ab data.tar.parte_ag data.tar.parte_al data.tar.parte_aq
data.tar.parte_ac data.tar.parte_ah data.tar.parte_am data.tar.parte_ar
data.tar.parte_ad data.tar.parte_ai data.tar.parte_an data.tar.parte_as
data.tar.parte_ae data.tar.parte_aj data_tar.parte_ao data.tar.parte_at
```

```
inf00047@odin-VirtualBox:~/cachos/partes$ cat ./data.tar.parte_* > data.tar.gz
```

Posteriormente tendríamos que desempaquetar los archivos descargados.

```
tar -xvf <paquete> -C <directorio_de_destino>
```

Este comando nos desempaquetaría el archivo en el directorio que queramos.

El siguiente paso sería instalar los servicios que necesitemos, una vez instalados, paramos el servicio y remplazamos los archivos de configuración creados por los que teníamos guardados de nuestras copias de seguridad y reiniciar el servicio. Hacemos lo mismo con los archivos de datos y si no hay conflicto de permisos ni nada por el estilo ya estaría restaurado. (en caso de las bases de datos tendríamos que insertar los archivos .sql)