

- يمكن استخدامها :-

- ### الاستخدام البسيط :

\* يتطلب صلاحيات مدير نظام.

```
$ tree tests/
tests/
├── 01
│   └── 4.txt
├── 1.txt
├── 2.txt
└── 3.txt
```

1 directory, 4 files

• نلاحظ أنه يوجد مجلد فرعي اسمه 01 .

سنقوم بمزامنه المحتوى الى مجلد **backups** .

```
$ rsync -v tests/* backups/
skipping directory 01
1.txt
2.txt
3.txt
```

```
sent 197 bytes received 73 bytes 540.00 bytes/sec
total size is 0 speedup is 0.00
```

كما تلاحظ من المخرج تم تخطي المجلد 01 و بالنظر إلى المجلد **backups** نرى انه يحتوي فقط على

1.txt , 2.txt , 3.txt .

```
$ tree backups/
backups/
├── 1.txt
├── 2.txt
└── 3.txt
```

0 directories, 3 files

لتفادي هذا السلوك الغير متوقع نستخدم خيار **r-** .

مثال:

```
$ rsync -rv tests backups/
sending incremental file list
tests/
tests/1.txt
tests/2.txt
tests/3.txt
tests/01/
tests/01/4.txt
```

sent 344 bytes received 108 bytes 904.00 bytes/sec  
total size is 0 speedup is 0.00

: محتوى backups

```
$ tree backups/  
backups/  
├── tests  
│   ├── 01  
│   │   └── 4.txt  
│   ├── 1.txt  
│   ├── 2.txt  
│   └── 3.txt
```

2 directories, 4 files

ملاحظة هامة :

إذا ابقيت علامة / بعد اسم المجلد - source/ - ستقوم بمزامنه المحتوى فقط , ولكن اذا حذفت العلامة source ستقوم بمزامنه كامل المجلد كما شاهدنا في المثال اعلاه tests بالكامل تمت مزامنته في backups .

قد ترغب بمزامنه المحتوى بدلا من مزامنه المجلد بالكامل في هذه الحالة ستضيف / إلى اسم المجلد.

```
$ rsync -rv tests/ backups/  
sending incremental file list  
1.txt  
2.txt  
3.txt  
01/  
01/4.txt
```

sent 326 bytes received 104 bytes 860.00 bytes/sec  
total size is 0 speedup is 0.00

```
$ tree backups/  
backups/  
├── 01  
│   └── 4.txt  
├── 1.txt  
├── 2.txt  
└── 3.txt
```

1 directory, 4 files

كما تشاهد تمت مزامنه محتوى مجلد tests بدلا من مزامنه كامل المجلد.

نصيحة مجرب:

تستطيع استخدام الخاصية `dry-run--` أو إختصاراً `n-` قبل تنفيذ عملية المزامنه , وستقوم بعرض المحتوى الذي سيتم مزامنته. هذه الخاصية مفيدة جداً بحيث أنها لاتقوم بعملية المزامنه الفعلية ولكن ستقوم بعرض المحتوى الذي سيتم مزامنته.

ملاحظة:

تحتاج إلى `verbose` لكي تعمل.

مثال:

```
$ rsync -rvn tests/ backups/
sending incremental file list
1.txt
2.txt
3.txt
01/
01/4.txt

sent 174 bytes received 32 bytes 412.00 bytes/sec
total size is 0 speedup is 0.00 (DRY RUN)
```

الملفات المعروضة هي التي سيتم مزامنتها. مجرد ماترى أن الأمور جيدة بالنسبة لك تستطيع حذف الخيار `n-` وتقوم بعملية المزامنه.

فرضاً لديك ملف اعدادات `configuration file` أو اي ملف تريد نقله والمحافظة على صلاحياته والملكية, إلخ....

في هذه الحالة ستضيف الخيار `a-` حيث أنه سيتم نقل الملفات في وضع الأرشفه مع المحافظة على الإختصارات و الأجهزة والخصائص والصلاحيات والملكية.

The files are transferred in archive mode, which ensures that symbolic links, devices, attributes, permissions, ownerships, etc. are preserved in the transfer.

مقارنة بسيطة:

بدون الخيار `a-`:

```
$ ls -la tests/
total 12
drwxr-xr-x 3 rakan rakan 4096 Jun  7 11:48 .
drwxr-xr-x 4 rakan rakan 4096 Jun  8 09:01 ..
drwxr-xr-x 2 rakan rakan 4096 Jun  7 11:48 01
-rwxrwxrwx 1 rakan rakan   0 Jun  7 11:48 1.txt
-rw-r--r-- 1 rakan rakan   0 Jun  7 11:48 2.txt
-rw-r--r-- 1 rakan rakan   0 Jun  7 11:48 3.txt
```

هنا قمنا بتغيير صلاحيات الملف `1.txt` إلى `777` فقط لتوضيح النقطة. وقمت بنسخ المحتوى باستخدام الأمر التالي:

```
rsync -rv tests/ backups/
```

الآن لو قمنا بالتحقق من الصلاحيات:

```
$ ls -la backups/
total 12
drwxr-xr-x 3 rakan rakan 4096 Jun  8 09:03 .
drwxr-xr-x 4 rakan rakan 4096 Jun  8 09:04 ..
drwxr-xr-x 2 rakan rakan 4096 Jun  8 09:03 01
-rwxr-xr-x 1 rakan rakan  0 Jun  8 09:03 1.txt
-rw-r--r-- 1 rakan rakan  0 Jun  8 09:03 2.txt
-rw-r--r-- 1 rakan rakan  0 Jun  8 09:03 3.txt
```

كما نرى تم تغيير الصلاحيات من 777 إلى 755 , لكن ماذا لو كنا راضين مع 777 .):

في هذه الحالة سنضيف الخيار **a** .

مع خيار **a**

لا نحتاج لإضافة الخيار **r**- بحيث أن الخيار **a** سيضيفه لنا.

```
rsync -av tests/ backups/
```

```
$ ls -la backups/
total 12
drwxr-xr-x 3 rakan rakan 4096 Jun  7 11:48 .
drwxr-xr-x 4 rakan rakan 4096 Jun  8 09:08 ..
drwxr-xr-x 2 rakan rakan 4096 Jun  7 11:48 01
-rwxrwxrwx 1 rakan rakan  0 Jun  7 11:48 1.txt
-rw-r--r-- 1 rakan rakan  0 Jun  7 11:48 2.txt
-rw-r--r-- 1 rakan rakan  0 Jun  7 11:48 3.txt
```

الآن حصلنا على المخرج المطلوب ونحن راضين ;)

فرضا قمنا بتعديل/تغيير ملف أو عدة ملفات, من الغير العملي بأنك تقوم بإعادة مزامنه كل شيء بالكامل.

بدلا من ذلك سنقوم باستخدام الخيار **u**- لمزامنه/نسخ فقط التغييرات.

دعنا نأخذ مثالنا السابق:

```
$ tree tests/
tests/
├── 01
│   └── 4.txt
├── 1.txt
├── 2.txt
├── 3.txt
└── 4.txt
```

1 directory, 5 files

4.txt.

الآن دعنا نستخدم **u** ونقوم بالتأكد من المحتوى الذي سيتم مزامنته عن طريق **n** وبالتأكيد سنقوم بالمقارنة باستخدام الهاش **c** لنكون دقيقين (;

```
$ rsync -avunc tests/ backups/
sending incremental file list
./
4.txt

sent 293 bytes received 23 bytes 632.00 bytes/sec
total size is 0 speedup is 0.00 (DRY RUN)
```

كما ترى فقط الملف **txt.4** الذي سيتم مزامنته. الآن نستطيع حذف الخيار **n** ونقوم بعملية المزامنة.

ماذا لو كنا نريد مزامنه عدة ملفات؟

تستطيع عمل ذلك بعدة طرق ولكن سنشرح أسهل طريقة وهي تحدد الملفات المطلوب مزامنتها داخل **{ }** مفصولين بفاصله.

مثال:

```
rsync -av {tests/,/etc/passwd} backups/
```

النتيجة :

```
$ tree backups/
backups/
├── 01
│   └── 4.txt
├── 1.txt
├── 2.txt
├── 3.txt
├── 4.txt
└── passwd

1 directory, 6 files
```

مزامنه ملفات تنتهي بإمتدادات أو ملفات معينة وتجاهل أي شيء آخر.

في هذه الحالة سنقوم باستخدام الخيارين **include--** و **exclude--** لإتمام العملية.

فرضا لدينا الهرميه التاليه ونريد بمزامنه فقط الملفات التي تنتهي ب **pdf** ونتجاهل أي شيء آخر.

```
$ tree tests/
tests/
├── 01
│   └── 4.txt
└── book2.pdf
```

```
|— 1.txt
|— 2.txt
|— 3.txt
|— 4.txt
|— blah.pdf
|— book1.pdf
|— news.pdf
```

1 directory, 9 files

```
rsync -avm --include="*/" --include="*.pdf" --exclude="*" tests/ backups/
```

```
$ tree backups/
backups/
|— 01
|   |— book2.pdf
|— blah.pdf
|— book1.pdf
|— news.pdf
```

1 directory, 4 files

حصلنا على النتيجة المطلوبة والآن سنقوم بتحليل الأمر الذي تم تنفيذه بالتفصيل:

- أولاً , `avm-` عدة خيارات معروفة مسبقاً لديك , الخيار `m-` يستخدم لتجنب التشعب داخل المجلدات الفارغة.
- ثانياً , `--include="*/"` تعني اذهب إلى المجلدات الفرعية في المجلد المراد النسخ منه.
- ثالثاً , `--include="*.pdf"` تعني قم بتضمين جميع الملفات التي تتضمن إمتداد `pdf`.
- أخيراً , `--exclude="*"` قم بتجاهل أي ملف لا يحتوي على الإمتداد `pdf`.

## عن بعد Remotely:

مثال : نسخ ملف من جهازك إلى سيرفر :

**يرجى الإنتباه:**

خدمة `ssh/rsync` لابد ان تكون تعمل على السيرفر أو على جهازك الشخصي في حالة النقل من السيرفر إلى جهازك.

```
rsync -v localFile user@10.10.10.10:~/remoteFile
```

**ملاحظة :**

إذا لم تقم بتحديد موقع على السيرفر سيتم وضع المحتوى داخل مجلد المنزل الخاص بالمستخدم.

مثال آخر:

```
rsync -v localFile user@10.10.10.10:
```

لم نقم تحديد موقع على السيرفر , اذا يتواجد الملف في /home/user/

مثال: نسخ ملف من السيرفر إلى جهازك الشخصي:

```
rsync -v user@10.10.10.10:~/remoteFile localFile
```

مثال : نسخ ملف من سيرفر مع رقم منفذ مخصص:

```
rsync -e 'ssh -p 1337' user@10.10.10.10:remoteFile localFile
```

مثال : نسخ كامل محتوى المجلد مع تفعيل خيار P- فقط الملفات الحديثة -u

```
rsync -urvP dirLocal/ user@10.10.10.10:~/backups/
```

نسخ عدة ملفات من السيرفر

```
rsync -zavP user@10.10.10.10:{/etc/passwd,/etc/hostname,/var/www/html/index.php} dirLocal/
```

### ملاحظة :

إستخدمنا -z لضغط الملفات في عملية النقل للتسريع وتقليل إستخدام البيانات.

### خاتمة:

rsync

أداة جدا قوية ويمكن إستخدامها لعدة استخدامات , كما انصح بالتعمق أكثر بالأداة , أستخدامها بعمليات الأتمتة , والعمليات المجدولة, إلخ...

لو كان عندك أي إستفسار خذ راحتك وتواصل معي على تويتر  ;

EOF