

Shockware Gaming S.A. | Crédito de Síntesi 2014

Rsync

rsync es una poderosa implementación de un pequeño y maravilloso algoritmo. Su principal poder es la habilidad de replicar eficientemente un sistema de archivos. Usando rsync, es fácil configurar un sistema que mantendrá una copia actualizada de un sistema de archivos usando un arreglo flexible de protocolos de red, tales como NFS, SMB o SSH. La segunda gran característica que consigue este sistema de copias de seguridad es la capacidad de archivar viejas copias de archivos que han sido modificadas o eliminadas.



“Descripción del servicio”

Instalación y configuración “del servicio”

Primeramente instalamos el servicio del Rsync en nuestro

ejecutamos `sudo apt-get install rsync`

Luego para hacer el Rsync debemos realizar el siguiente comando

- avz: este parametro se refiere a todos los archivo de la transferencia de un ordenador a otro
- progress: muestra la información de progreso de la transferencia
- Stats: Muestra la estadísticas sobre la transferencia de archivos
- e: Esta es una opcion es para preservar la ejecutabilidad de los archivos
- delete: elimina los archivos extraños del dest dirs

usuario@IP_DEL_ORIGEN:/Direccion_de_los_archivos /destino_de_los_archivo

*Con la siguiente orden estamos realizando una copia de seguridad del home de Shockware-Gaming a nuestro raid

```
rsync -avzrp --delete --stats --progress -e ssh  
shockgadmin@192.168.19.233:/home/Shockware-Gaming /backup
```

*Con la siguiente orden estamos realizando una copia de seguridad del directorio opt a nuestro raid

```
rsync -avzrp --delete --stats --progress -e ssh shockgadmin@192.168.19.233:/opt/tomcat*  
/backup
```

*Con la siguiente orden estamos realizando una copia de seguridad del directorio var www a nuestro raid

```
rsync -avzrp --delete --stats --progress -e ssh shockgadmin@192.168.19.233:/var/www  
/backup
```

Crontab

Crontab es un archivo de texto que guarda una lista de comandos o comando a ejecutar en un tiempo especificado por el usuario. El Crontab verificará la fecha y hora en que se debe ejecutar el script o el comando. Donde cada usuario puede tener su propio archivo crontab, de hecho el /etc/crontab se asume que es el archivo crontab del usuario root, cuando los usuarios normales (e incluso root) desean generar su propio archivo de crontab, entonces utilizaremos el comando crontab.

Nosotros utilizaremos un crontab para que de esta manera se pueda realizarse automáticamente la copia de seguridad de nuestro servidor 1 cada determinado tiempo

Primeramente crearemos un script para que de esta manera el crontab lo ejecute

```
#!/bin/bash

echo "Realizando Copia";

rsync -avzrp --delete --stats --progress -e ssh shockgadmin@192.168.19.233:/home/
Shockware-Gaming /backup

rsync -avzrp --delete --stats --progress -e ssh shockgadmin@192.168.19.233:/opt/
tomcat* /backup

rsync -avzrp --delete --stats --progress -e ssh shockgadmin@192.168.19.233:/var/
www /backup

rsync -avzrp --delete --stats --progress -e ssh shockgadmin@192.168.19.233:/etc /
backup

then

echo "BACKUP OK"

else

echo "BACKUP FAILED"
```

Guardamos el script en →

Abrimos el crontab

ejecutamos crontab -e luego seleccionamos el editor de texto

```
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
0 21 * * 1,3,4 /home/shockgadmin/.scripts/rsync.sh && echo "Copia realizada" > $HOME/rsync.log ; date >> $HOME/rsync.log
45 17 * * 2 /home/shockgadmin/.scripts/rsync.sh && echo "Copia realizada" > $HOME/rsync.log ; date >> $HOME/rsync.log
10 19 * * 5 /home/shockgadmin/.scripts/rsync.sh && echo "Copia realizada" > $HOME/rsync.log ; date >> $HOME/rsync.log
```

Ponemos que el contrab se ejecute los días lunes,miércoles y jueves alas 9pm y que cree un archivo en la cual lo guardara en la carpeta de donde se encuentra el scripts, hacemos otra la misma orden para que se ejecute los días martes y viernes.

Como usuario SHOCKGADMIN no podemos realizar las copias de seguridad puesto que algunos ficheros remotos son del usuario ROOT. Necesitamos hacer RSYNC transparente también con el USUARIO ROOT.

```
etc/speech-dispatcher/
etc/speech-dispatcher/clients/
etc/speech-dispatcher/modules/
etc/ssh/
rsync: send_files failed to open "/etc/ssh/ssh_host_dsa_key": Permission denied (13)
rsync: send_files failed to open "/etc/ssh/ssh_host_ecdsa_key": Permission denied (13)
rsync: send_files failed to open "/etc/ssh/ssh_host_rsa_key": Permission denied (13)
etc/ssl/
etc/ssl/certs/
rsync: send_files failed to open "/etc/ssl/certs/shock2.pem": Permission denied (13)
etc/ssl/certs/java/
etc/ssl/private/
etc/sudoers.d/
rsync: send_files failed to open "/etc/sudoers.d/README": Permission denied (13)
etc/sysctl.d/
etc/systemd/
etc/systemd/system/
etc/systemd/system/multi-user.target.wants/
etc/terminfo/
etc/thunderbird/
etc/udev/
etc/udev/rules.d/
etc/ufw/
rsync: send_files failed to open "/etc/ufw/after.rules": Permission denied (13)
rsync: send_files failed to open "/etc/ufw/after6.rules": Permission denied (13)
```

Sudo ssh-keygen -tdsa



```
scp id_dsa.pub root@192.168.19.233:/root/.ssh/authorized_keys_root
```