

# Automate Backup in Linux Using RSYNC & Crontab | Backup Data On Remote Server Using RSYNC

rsync or remote synchronization is a software utility for Unix-Like systems that efficiently sync files and directories between two hosts or machines.

One of them being the source or the local-host from which the files will be synced, the other one being the remote-host, on which synchronization will take place.

There are basically two ways in which rsync can copy/sync data:

Copying/syncing to/from another host over any remote shell like ssh, rsh.

Copying/Syncing through rsync daemon using TCP.

=====

```
yum repolist
```

```
rpm -qa | grep rsync
```

```
yum install rsync -y
```

```
rpm -qi rsync
```

```
su - vikasnehra
```

```
mkdir backup
```

```
cd backup
```

```
touch test{1..10}.txt
```

```
mkdir -p /tmp/backup
```

```
rsync --help
```

```
rsync -v /home/vikasnehra/backup/* /tmp/backup/
```

```
exit
```

```
mkdir -p /backupsources/
```

```
cd /backupsources/ ; touch test{1..10}.txt
```

```
CLIENT: (192.168.1.108)
```

```
yum repolist
```

```
rpm -qa | grep rsync
```

```
yum install rsync -y
```

```
mkdir -p /backupdestination/
```

Go To SERVER:

```
rsync -av -e ssh /backupsources/ root@192.168.1.108:/backupdestination/
```

Go To CLIENT:

```
cd /backupdestination/
```

```
ls -lha
```

```
rm -rf *
```

```
ls -lha /root/.ssh/
```

```
mkdir -p /root/.ssh/
```

```
chmod 700 /root/.ssh/
```

```
touch /root/.ssh/authorized_keys
```

Go To SERVER:

**ssh-keygen**

**scp -r /root/.ssh/id\_rsa.pub root@192.168.1.108:/root/.ssh/authorized\_keys**

**date**

**crontab -e**

25 20 \* \* \* bash /root/backup.sh

**vim /root/backup.sh**

#!/bin/bash

/usr/bin/rsync -av -e ssh /backupsources/ root@192.168.1.108:/backupdestination/

**crontab -l**

**tail -f /var/log/cron**

Go To CLIENT:

**ll /backupdestination/**

===

#### # Client

```
[root@client ~]# uname -a
Linux client 4.18.0-80.el8.x86_64 #1 SMP Wed Mar 13 12:02:46 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
[root@client ~]#
```

#### #Server

```
[root@server ~]# #Welcome to Nehra Classes Youtube Channel#
[root@server ~]# #Automate Backup Using rsync & crontab in Linux#
[root@server ~]# uname -a
Linux server 3.10.0-1062.el7.x86_64 #1 SMP Thu Jul 18 20:25:13 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
[root@server ~]#
```

```
[root@server ~]# yum repolist
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
repo id      repo name                                          status
*epel/x86_64 Extra Packages for Enterprise Linux 7 - x86_64 13,322
localrepo    localrepo                                         5,229
remi-php72   Remi's PHP 7.2 RPM repository for Enterprise Linux 7 - x86_64 407
remi-safe    Safe Remi's RPM repository for Enterprise Linux 7 - x86_64 3,809
repolist: 22,767
[root@server ~]#
```

```
[root@server ~]# rpm -qa | grep rsync
rsync-3.1.2-6.el7_6.1.x86_64
```

#### # 1. Rsync backup inside same machine ( folder to Folder)

```
[root@server ~]# su - vikasnehra
Last login: Sun Jun 14 15:34:59 IST 2020 on pts/1
[vikasnehra@server ~]$ ll
total 4
drwxrwxr-x. 2 vikasnehra vikasnehra 192 Jun 14 15:35 backup
-rw-r--r--. 1 root      root      211 Jun  4 19:43 httpd.conf
drwx-----. 8 vikasnehra vikasnehra 267 May 11 11:23 Maildir
[vikasnehra@server ~]$ cd backup/
[vikasnehra@server backup]$ ll
```

```
[vikasnehra@server backup]$ ll
total 4
-rw-rw-r--. 1 vikasnehra vikasnehra 136 Jun 14 15:35 cal.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test10.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test1.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test2.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test3.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test4.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test5.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test6.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test7.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test8.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:35 test9.txt
[vikasnehra@server backup]$
```

```
[vikasnehra@server backup]$ rm -rf *
[vikasnehra@server backup]$ ll
total 0
[vikasnehra@server backup]$
```

REGISTERED VERSION - Please support MohaXterm by subscribing to the professional edition here: <https://mohaxterm.mohatek.net>

```
[vikasnehra@server backup]$ touch test{1..10}.txt
[vikasnehra@server backup]$ ll
total 0
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test10.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test1.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test2.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test3.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test4.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test5.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test6.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test7.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test8.txt
-rw-rw-r--. 1 vikasnehra vikasnehra 0 Jun 14 19:56 test9.txt
[vikasnehra@server backup]$ cal > cal.txt
[vikasnehra@server backup]$ ll
```



```

[vikasnehra@server backup]$ cd /tmp
[vikasnehra@server tmp]$ mkdir -p /tmp/backup
[vikasnehra@server tmp]$ cd backup/
[vikasnehra@server backup]$ ll
total 4
-rw-rw-r--. 1 vikasnehra vikasnehra 136 Jun 14 15:10 cal.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test10.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test1.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test2.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test3.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test4.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test5.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test6.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test7.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test8.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test9.txt
[vikasnehra@server backup]$ rm -rf *
[vikasnehra@server backup]$ █

[vikasnehra@server backup]$ ll
total 4
-rw-rw-r--. 1 vikasnehra vikasnehra 136 Jun 14 19:57 cal.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test10.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test1.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test2.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test3.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test4.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test5.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test6.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test7.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test8.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 19:56 test9.txt
[vikasnehra@server backup]$ █

```

#### # Create a/ tmp Dir

```

[vikasnehra@server backup]$ cd /tmp
[vikasnehra@server tmp]$ mkdir -p /tmp/backup
[vikasnehra@server tmp]$ cd backup/
[vikasnehra@server backup]$ ll
total 4
-rw-rw-r--. 1 vikasnehra vikasnehra 136 Jun 14 15:10 cal.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test10.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test1.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test2.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test3.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test4.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test5.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test6.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test7.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test8.txt
-rw-rw-r--. 1 vikasnehra vikasnehra  0 Jun 14 15:10 test9.txt
[vikasnehra@server backup]$ rm -rf *
[vikasnehra@server backup]$ █

```

```
[vikasnehra@server backup]$ rm -rf *
[vikasnehra@server backup]$ ll
total 0
[vikasnehra@server backup]$ clear
```

REGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

```
vikasnehra@server backup]$ rsync --he
```

```
or rsync [OPTION]... [USER@]HOST::SRC [DEST]
or rsync [OPTION]... rsync://[USER@]HOST[:PORT]/SRC [DEST]
The ':' usages connect via remote shell, while '::' & 'rsync://' usages connect
to an rsync daemon, and require SRC or DEST to start with a module name.
```

#### Options

-v, --verbose	increase verbosity
--info=FLAGS	fine-grained <b>informational</b> verbosity
--debug=FLAGS	fine-grained debug verbosity
--msgs2stderr	special output handling for debugging
-q, --quiet	suppress non-error messages
--no-motd	suppress daemon-mode MOTD (see manpage caveat)
-c, --checksum	skip based on checksum, not mod-time & size
-a, --archive	archive mode; equals -rlptgoD ( <b>no</b> -H,-A,-X)
--no-OPTION	turn off an implied OPTION (e.g. --no-D)
-r, --recursive	recurse into directories
-R, --relative	use relative path names
--no-implied-dirs	don't send implied dirs with --relative
-b, --backup	make backups (see --suffix & --backup-dir)

#### # Create a backup using Rsync

```
[vikasnehra@server backup]$ cd
[vikasnehra@server ~]$ rsync -v /home/vikasnehra/backup/* /tmp/backup/
cal.txt
test1.txt
test10.txt
test2.txt
test3.txt
test4.txt
test5.txt
test6.txt
test7.txt
test8.txt
test9.txt

sent 788 bytes  received 225 bytes  2,026.00 bytes/sec
total size is 136  speedup is 0.13
[vikasnehra@server ~]$
```



```
vikasnehra@server ~]$ cd /tmp/backup/
vikasnehra@server backup]$ ll
total 4
-rw-rw-r--. 1 vikasnehra vikasnehra 136 Jun 14 19:59 cal.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test10.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test1.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test2.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test3.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test4.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test5.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test6.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test7.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test8.txt
-rw-rw-r--. 1 vikasnehra vikasnehra   0 Jun 14 19:59 test9.txt
vikasnehra@server backup]$
```

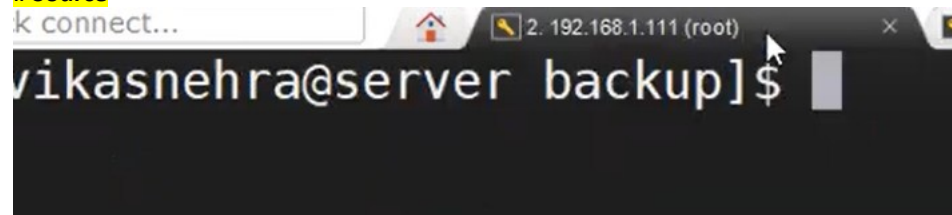
```
[vikasnehra@server backup]$ cat cal.txt
      June 2020
Su Mo Tu We Th Fr Sa
      1  2  3  4  5  6
 7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30
```

```
[vikasnehra@server backup]$ c
```

# Files have successfully created /tmp folder -in same Machine

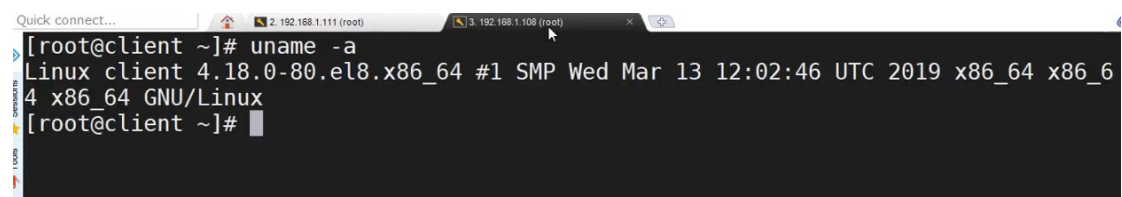
# 2 . Rsync backup Source to Destination machine

# Source



```
Quick connect... 2. 192.168.1.111 (root)
vikasnehra@server backup]$
```

#Destination - Remote location



```
Quick connect... 3. 192.168.1.108 (root)
[root@client ~]# uname -a
Linux client 4.18.0-80.el8.x86_64 #1 SMP Wed Mar 13 12:02:46 UTC 2019 x86_64 x86_6
4 x86_64 GNU/Linux
[root@client ~]#
```

```
192.168.1.108 (root)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MUXExec Tunneling Packages Settings Help
Quick connect... 2. 192.168.1.111 (root) 3. 192.168.1.108 (root)

[root@client ~]# ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.108 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::8a34:33b1:f182:2534 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:69:12:bd txqueuelen 1000 (Ethernet)
    RX packets 1938 bytes 322888 (315.3 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 116 bytes 21476 (20.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 48 bytes 5616 (5.4 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 48 bytes 5616 (5.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```

```
Quick connect... 2. 192.168.1.111 (root) 3. 192.168.1.108 (root)

[vikasnehra@server backup]$ ping 192.168.1.108
PING 192.168.1.108 (192.168.1.108) 56(84) bytes of data.
64 bytes from 192.168.1.108: icmp_seq=1 ttl=64 time=0.583 ms
64 bytes from 192.168.1.108: icmp_seq=2 ttl=64 time=0.390 ms
```

**# Come back to Root user account- Source**

```
Quick connect... 2. 192.168.1.111 (root) 3. 192.168.1.108 (root)

[vikasnehra@server backup]$ logout
[root@server ~]#
```

```
Quick connect... 2. 192.168.1.111 (root) 3. 192.168.1.108 (root)

[root@server ~]# mkdir /backupsources/
[root@server ~]# cd /backupsources/
[root@server backupsources]# ll
total 0
[root@server backupsources]# touch test{1..20}.txt
[root@server backupsources]# cal > cal.txt
[root@server backupsources]#
```



```
rw-r--r--. 1 root root 0 Jun 14 20:01 test12.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test13.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test14.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test15.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test16.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test17.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test18.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test19.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test1.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test20.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test2.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test3.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test4.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test5.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test6.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test7.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test8.txt
rw-r--r--. 1 root root 0 Jun 14 20:01 test9.txt
[root@server backupsource]#
```

#### # Go to Destination ... Create a backup

```
[root@client ~]# rpm -qa | grep rsync
rsync-3.1.3-4.el8.x86_64
[root@client ~]# clear
```

```
[root@client ~]# mkdir -p /backupdestination/
[root@client ~]# cd /backupdestination/
[root@client backupdestination]# ll
total 0
[root@client backupdestination]#
```

#### # Now time to Rsync from Source to Destination Move the backupfiles to Remote location

```
[root@server backupsource]# rsync -av -e ssh /backupsource/ root@192.168.1.108:/ba
ckupdestination/
The authenticity of host '192.168.1.108 (192.168.1.108)' can't be established.
ECDSA key fingerprint is SHA256:9kj2akycdfmtziFmDcWVrZHAsKYjqvuYNaiqcYr55h0.
ECDSA key fingerprint is MD5:5b:27:c6:bd:ae:c3:47:8d:1c:c7:df:f1:4a:1e:2a:9d.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.108' (ECDSA) to the list of known hosts.
root@192.168.1.108's password: █
```

```
test14.txt
test15.txt
test16.txt
test17.txt
test18.txt
test19.txt
test2.txt
test20.txt
test3.txt
test4.txt
test5.txt
test6.txt
test7.txt
test8.txt
test9.txt

sent 1,379 bytes  received 418 bytes  133.11 bytes/sec
total size is 136  speedup is 0.08
[root@server backupsource]# █
```

#### # Check Destination Server

```
[root@client ~]# mkdir -p /backupdestination/
[root@client ~]# cd /backupdestination/
[root@client backupdestination]# ll
total 0
[root@client backupdestination]# ll █
```





## # Source

```
Quick connect... 2. 192.168.1.111 (root) 3. 192.168.1.108 (root)
42 01 * * * bash /root/backup.sh
```

42 : min / 01 -hr / \* -day off month / \* - each month / \* day of Week-each day /script-Location

## # Save + exit : wq!

```
Quick connect... 2. 192.168.1.111 (root) 3. 192.168.1.108 (root)
[root@server backupsource]# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
[root@server backupsource]# crontab -l
42 01 * * * bash /root/backup.sh
[root@server backupsource]#
```

## # Create Script file for this crontab , which will automate the backup process

```
root@server backupsource]# vim /root/backup.sh
#!/bin/bash
/usr/bin/rsync -av -e ssh /backupsources/ root@192.168.1.108:/backupdestination/
```

## # date cmd backup source server

```
[root@server backupsource]# date
Sun Jun 14 20:36:49 IST 2020
[root@server backupsource]#
```

```
[root@server backupsource]# crontab -e
46 20 * * * bash /root/backup.sh
```

This script will execute @ 20:hrs: 46mins in this machine

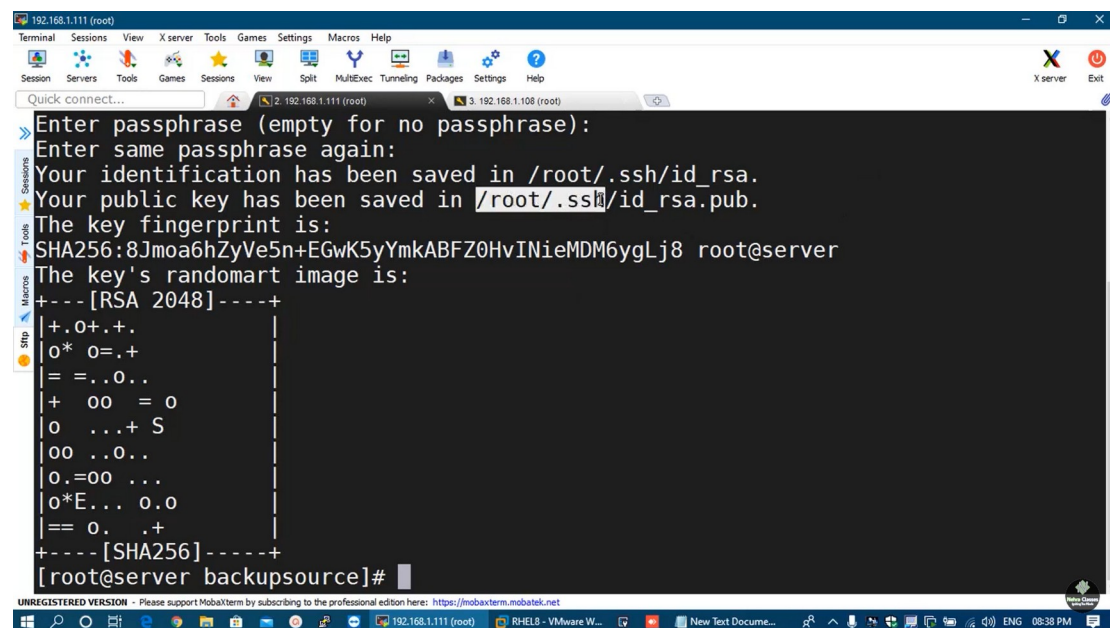
# SaVE + EXIT - wq!

```
[root@server backupsource]# crontab -e  
crontab: installing new crontab  
[root@server backupsource]#
```

# move to destination sever

```
[root@client backupdestination]# ll  
total 0  
[root@client backupdestination]#
```

# Create ssh-keys in Source



The screenshot shows a MobaXterm terminal window with a menu bar (Terminal, Sessions, View, X server, Tools, Games, Settings, Macros, Help) and a toolbar. The terminal displays the output of the 'ssh-keygen' command. The user is prompted to enter a passphrase (empty for no passphrase) and to re-enter the same passphrase. The output indicates that the identification has been saved in /root/.ssh/id\_rsa, the public key has been saved in /root/.ssh/id\_rsa.pub, and the key fingerprint is SHA256:8Jmoa6hZyVe5n+EGwK5yYmkABFZ0HvINieMDM6ygLj8. The key's randomart image is also displayed as a series of characters. The terminal prompt is [root@server backupsource]#.

```
192.168.1.111 (root)  
Terminal Sessions View X server Tools Games Settings Macros Help  
Session Servers Tools Games Sessions View Split MUXExec Tunneling Packages Settings Help  
Quick connect... 192.168.1.111 (root) 192.168.1.108 (root)  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /root/.ssh/id_rsa.  
Your public key has been saved in /root/.ssh/id_rsa.pub.  
The key fingerprint is:  
SHA256:8Jmoa6hZyVe5n+EGwK5yYmkABFZ0HvINieMDM6ygLj8 root@server  
The key's randomart image is:  
+---[RSA 2048]---+  
|.o+..+|  
o* o=.+|  
|=..o..|  
+ oo = o|  
o ...+ S|  
oo ..o..|  
o.=oo ...|  
o*E... o.o|  
|= o. .+|  
+---[SHA256]-----+  
[root@server backupsource]#
```

# Go to server set the below cmd

```
[root@client backupdestination]# mkdir -p /root/.ssh/  
[root@client backupdestination]#
```

# Source to destination Server

```
[root@server backupsource]# scp -r /root/.ssh/id_rsa.pub root@192.168.1.108:/root/.ssh/authorized_keys
root@192.168.1.108's password:
id_rsa.pub                                100% 393   381.6KB/s   00:00
[root@server backupsource]#
```

# Modify the permissions on the destination server

```
[root@client backupdestination]# chmod 700 /root/.ssh/
[root@client backupdestination]# chmod 600 /root/.ssh/authorized_keys
```

# Source

```
[root@server backupsource]# tail -f /var/log/cron
Jun 14 20:34:01 server crond[1395]: (root) RELOAD (/var/spool/cron/root)
Jun 14 20:34:58 server crontab[3177]: (root) BEGIN EDIT (root)
Jun 14 20:35:13 server crontab[3177]: (root) REPLACE (root)
Jun 14 20:35:13 server crontab[3177]: (root) END EDIT (root)
Jun 14 20:36:01 server crond[1395]: (root) RELOAD (/var/spool/cron/root)
Jun 14 20:37:01 server crontab[3199]: (root) BEGIN EDIT (root)
Jun 14 20:37:15 server crontab[3199]: (root) REPLACE (root)
Jun 14 20:37:15 server crontab[3199]: (root) END EDIT (root)
Jun 14 20:38:01 server crond[1395]: (root) RELOAD (/var/spool/cron/root)
Jun 14 20:40:01 server CROND[3231]: (root) CMD (/usr/lib64/sa/sa1 1 1)
```

```
[root@server backupsource]# tail -f /var/log/cron
Jun 14 20:34:01 server crond[1395]: (root) RELOAD (/var/spool/cron/root)
Jun 14 20:34:58 server crontab[3177]: (root) BEGIN EDIT (root)
Jun 14 20:35:13 server crontab[3177]: (root) REPLACE (root)
Jun 14 20:35:13 server crontab[3177]: (root) END EDIT (root)
Jun 14 20:36:01 server crond[1395]: (root) RELOAD (/var/spool/cron/root)
Jun 14 20:37:01 server crontab[3199]: (root) BEGIN EDIT (root)
Jun 14 20:37:15 server crontab[3199]: (root) REPLACE (root)
Jun 14 20:37:15 server crontab[3199]: (root) END EDIT (root)
Jun 14 20:38:01 server crond[1395]: (root) RELOAD (/var/spool/cron/root)
Jun 14 20:40:01 server CROND[3231]: (root) CMD (/usr/lib64/sa/sa1 1 1)
Jun 14 20:46:01 server CROND[3296]: (root) CMD (bash /root/backup.sh)
```

# destination server

```
[root@client backupdestination]# ll
total 4
-rw-r--r--. 1 root root 136 Jun 14 20:01 cal.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test10.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test11.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test12.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test13.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test14.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test15.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test16.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test17.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test18.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test19.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test1.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test20.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test2.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test3.txt
-rw-r--r--. 1 root root  0 Jun 14 20:01 test4.txt
```

# Note : If I need to backup every min / every hr / every sec from Source to Destination machine



```
[root@server backupsource]# crontab -e  
crontab: installing new crontab  
[root@server backupsource]#
```

Click Connect...

2. 192.168.1.111 (root) x 3. 192.168.1.108 (root)

```
* * * * * bash /root/backup.sh
```