Backup & Recovery SPSE

Direktorat Pengembangan Sistem Pengadaan Secara Elektronik
LKPP

Introduction

What's a Backup?

Data Backup

[da·ta·back·up] **noun**

A copy or archive of your important information on a device.

The act of **backing up your data** is when you:

Create a copy of your important information.

Store it in a secure, separate location.

Recognize the backup as a restoration method for your device.

https://us.norton.com/internetsecurity-how-to-the-importance-of-data-back-up.html

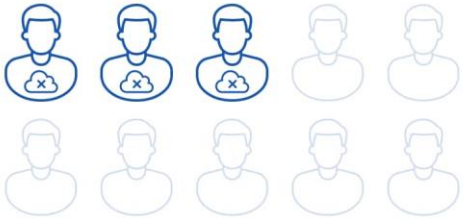
Introduction

Facts and Figures

Consider these statistics — and that backing up your data can put you ahead of losing it all.

1 in 10 computers are infected with viruses each month.

(World Backup Day)



30% of people have never backed up their device.

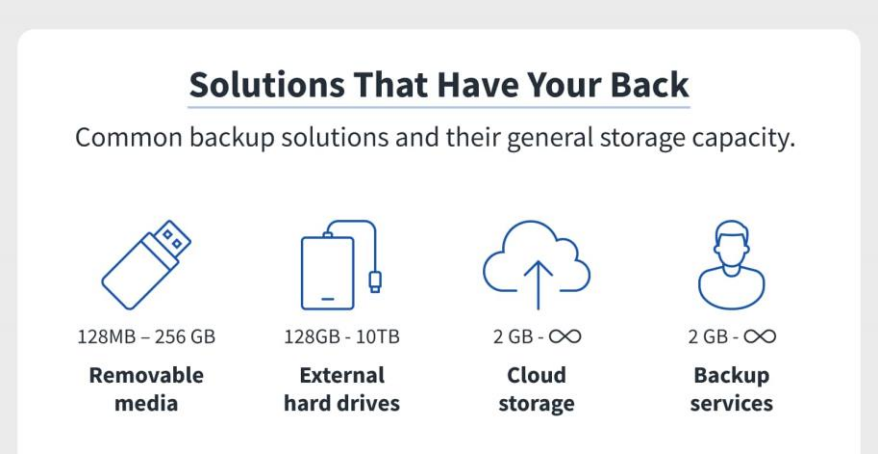
(World Backup Day)

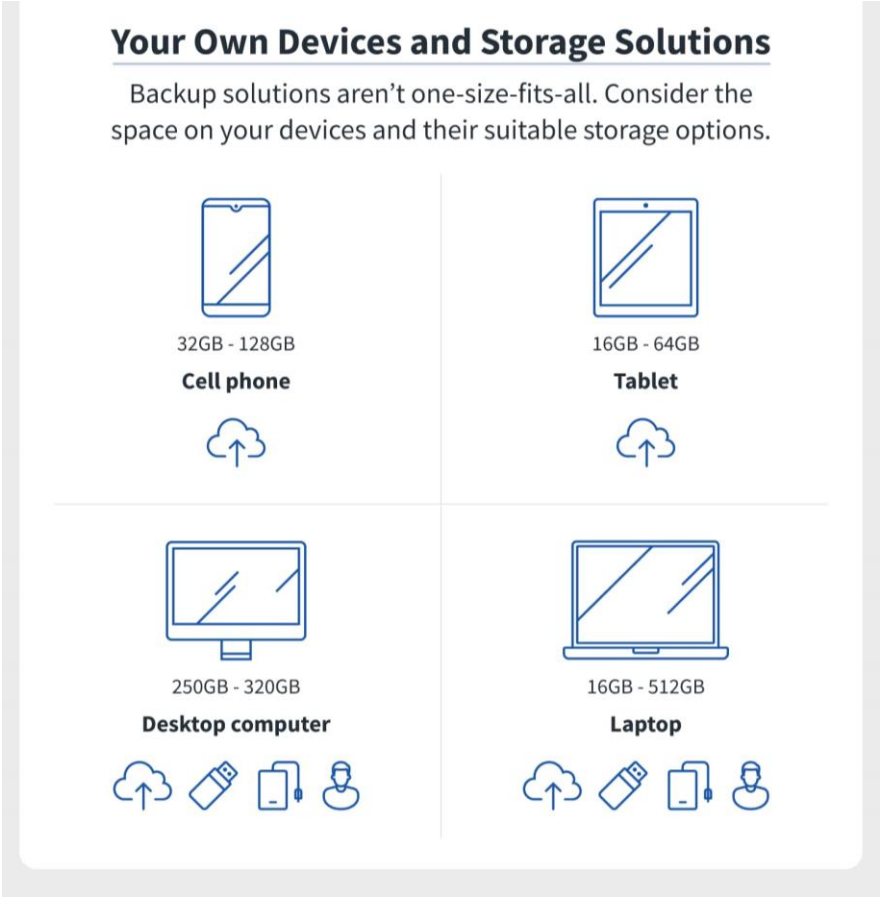
70 million cell phones are lost each year.

(Kensington)

https://us.norton.com/internetsecurity-how-to-the-importance-of-data-back-up.html

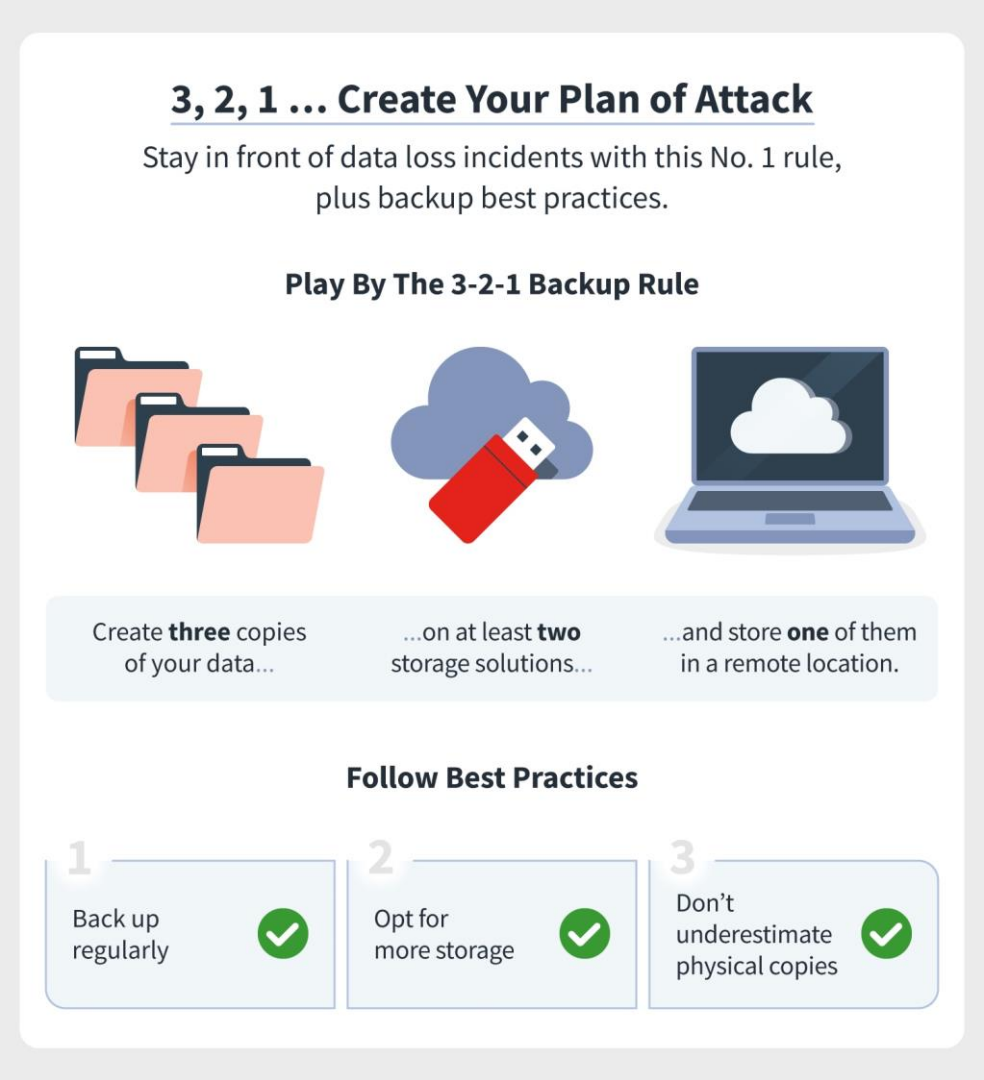
Introduction





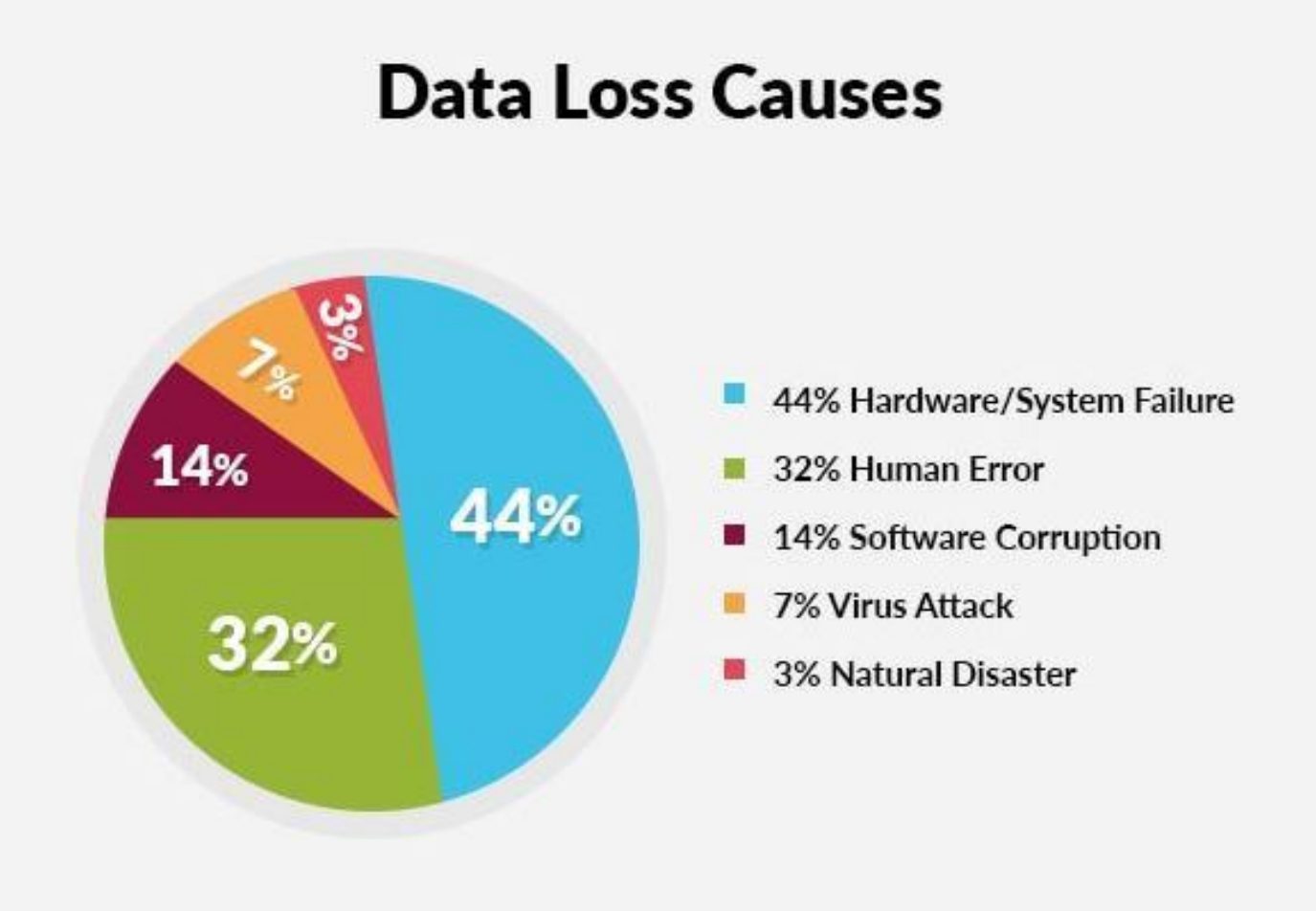
https://us.norton.com/internetsecurity-how-to-the-importance-of-data-back-up.html

Introduction



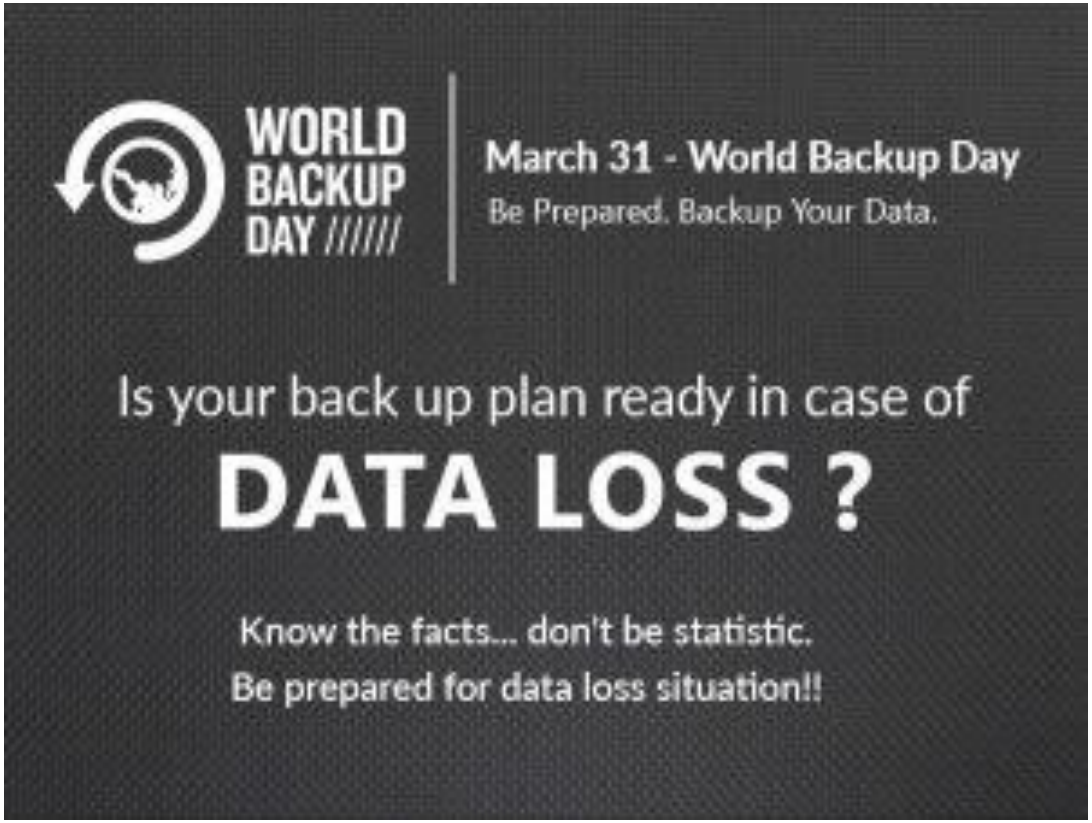
https://us.norton.com/internetsecurity-how-to-the-importance-of-data-back-up.html

Introduction



https://www.stellarinfo.co.in/blog/world-backup-day/

Introduction



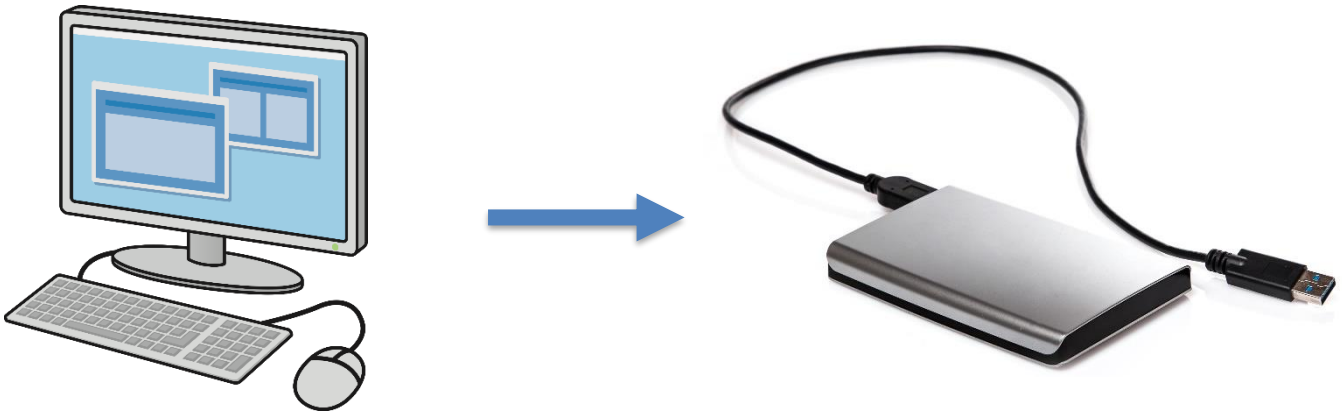
https://www.stellarinfo.co.in/blog/world-backup-day/

Backup & Recovery

	Prosedur	Aktivitas
1	B&R HDD External	1. Full backup (database) 2. Incremental (datafile upload/download)
2	B&R Server via SSH (remote)	1. Full backup (database) 2. Incremental (datafile upload/download)
3	B&R Server via NFS	1. Full backup (database) 2. Incremental (datafile upload/download)

(1) Backup & Recovery HDD Eksternal

Mounting HDD External / Flashdisk External



Step Mounting: check the path > do mounting > verify the mounting

Step Backup: create folder backup > backup database > synchronize file backup (DB and file_prod)

(1) Backup & Recovery HDD Eksternal

Mounting HDD External / Flashdisk External

1. Check the path of the inserted storage device

fdisk –l …. (find disk and list them)

```
root@localhost:~
[root@localhost ~]# fdisk -l

Disk /dev/sda: 8589 MB, 8589934592 bytes, 16777216 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000cdd94

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *        2048     2099199     1048576   83  Linux
/dev/sda2         2099200    16777215     7339008   8e  Linux LVM

Disk /dev/mapper/centos-root: 6652 MB, 6652166144 bytes, 12992512 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mapper/centos-swap: 859 MB, 859832320 bytes, 1679360 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

[root@localhost ~]#
```

Cont..

```
root@localhost:~
[root@localhost ~]# fdisk -l

Disk /dev/sda: 8589 MB, 8589934592 bytes, 16777216 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000cdd94

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *        2048     2099199     1048576   83  Linux
/dev/sda2         2099200    16777215     7339008   8e  Linux LVM

Disk /dev/mapper/centos-root: 6652 MB, 6652166144 bytes, 12992512 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mapper/centos-swap: 859 MB, 859832320 bytes, 1679360 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/sdb: 15.5 GB, 15527313408 bytes, 30326784 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1   *        2048    30326783    15162368    c  W95 FAT32 (LBA)
[root@localhost ~]#
```

(1) Backup & Recovery HDD Eksternal

Mounting HDD External / Flashdisk External

1. Check the path of the inserted storage device
fdisk –l …. (find disk and list them)

```
root@localhost:~
[root@localhost ~]# fdisk -l

Disk /dev/sda: 8589 MB, 8589934592 bytes, 16777216 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000cdd94

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *        2048     2099199     1048576   83  Linux
/dev/sda2         2099200    16777215     7339008   8e  Linux LVM

Disk /dev/mapper/centos-root: 6652 MB, 6652166144 bytes, 12992512 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mapper/centos-swap: 859 MB, 859832320 bytes, 1679360 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/sdb: 15.5 GB, 15527313408 bytes, 30326784 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1   *        2048    30326783    15162368    c  W95 FAT32 (LBA)
[root@localhost ~]#
```

Floppy disks: The floppy disks are recognized as the “fd” keyword, and if one floppy disk is detected, then it would be “fd1”; for a second, it would be “fd2” and so on for other floppy disks, “fd3”,… “fdn”.

Hard disks: Like the floppy disks, the hard disks are represented by “sda”, “sdb”, and so on. “sda” is used to represent the first hard disk detected, “sdb” for the second hard disk, and the sequence continues.

Cont..

(1) Backup & Recovery HDD Eksternal

Mounting HDD External / Flashdisk External

1. Check the path of the inserted storage device

sudo parted -l …. (list parted storage)

```
root@localhost:~
[root@localhost ~]# sudo parted -l
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sda: 8590MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  1075MB  1074MB  primary  xfs          boot
 2      1075MB  8590MB  7515MB  primary               lvm


Model: hp v212w (scsi)
Disk /dev/sdb: 15.5GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  15.5GB  15.5GB  primary  fat32        boot, lba


Model: Linux device-mapper (linear) (dm)
Disk /dev/mapper/centos-swap: 860MB
Sector size (logical/physical): 512B/512B
Partition Table: loop
Disk Flags:

Number  Start  End    Size   File system     Flags
 1      0.00B  860MB  860MB  linux-swap(v1)


Model: Linux device-mapper (linear) (dm)
Disk /dev/mapper/centos-root: 6652MB
Sector size (logical/physical): 512B/512B
Partition Table: loop
Disk Flags:

Number  Start  End     Size    File system  Flags
 1      0.00B  6652MB  6652MB  xfs
```

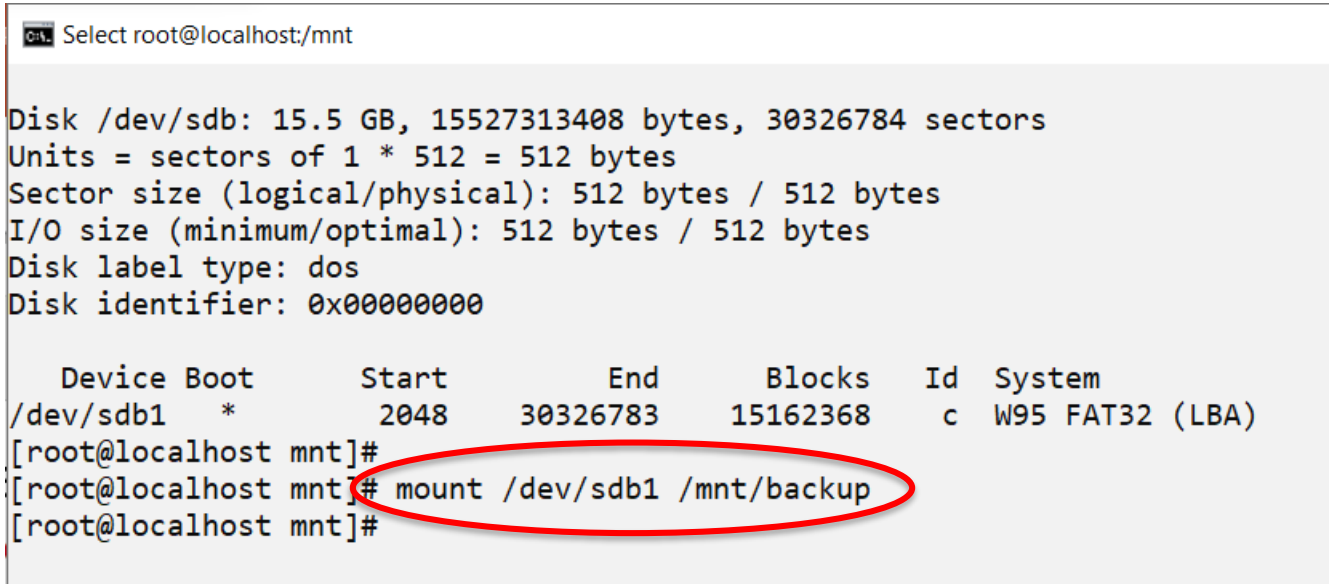
(1) Backup & Recovery HDD Eksternal

2. Mounting HDD External / Flashdisk External

mount –t ntfs /dev/sda.. /mnt/backup

mount –t ext4 /dev/sda.. /mnt/backup

mount /dev/sda… /mnt/backup



unmount /dev/sda… /mnt/backup

(1) Backup & Recovery HDD Eksternal

3. Verify the Mounting HDD External / Flashdisk External

lsbk

lists information about all or the specified block devices.

```
root@localhost:~
[root@localhost ~]#
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0    8G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0    7G  0 part
  ├─centos-root 253:0    0  6.2G  0 lvm  /
  └─centos-swap 253:1    0  820M  0 lvm  [SWAP]
sdb               8:16   1 14.5G  0 disk
└─sdb1            8:17   1 14.5G  0 part
sr0              11:0    1 1024M  0 rom
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# mount /dev/sdb1 /mnt/backup/
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0    8G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0    7G  0 part
  ├─centos-root 253:0    0  6.2G  0 lvm  /
  └─centos-swap 253:1    0  820M  0 lvm  [SWAP]
sdb               8:16   1 14.5G  0 disk
└─sdb1            8:17   1 14.5G  0 part /mnt/backup
sr0              11:0    1 1024M  0 rom
[root@localhost ~]# client_loop: send disconnect: Connection reset
```

(1) Backup & Recovery HDD Eksternal

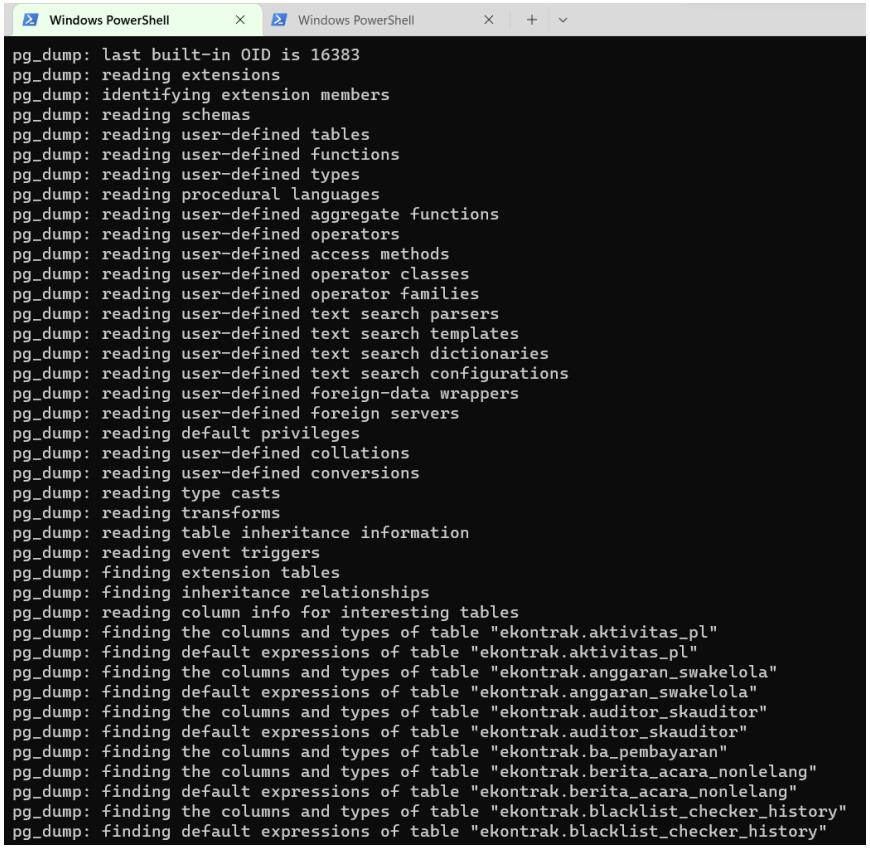
4. Create folder to save file backup

```
# mkdir /home/backup
# mkdir /mnt/backup/file_prod
# mkdir /mnt/backup/database
```

(1) Backup & Recovery HDD Eksternal

5. Backup database command

su postgres –c ‘pg_dump -F c -v -f '/home/backup/epns_prod_`data ‘+%d-%m-%Y’`.backup epns_prod’

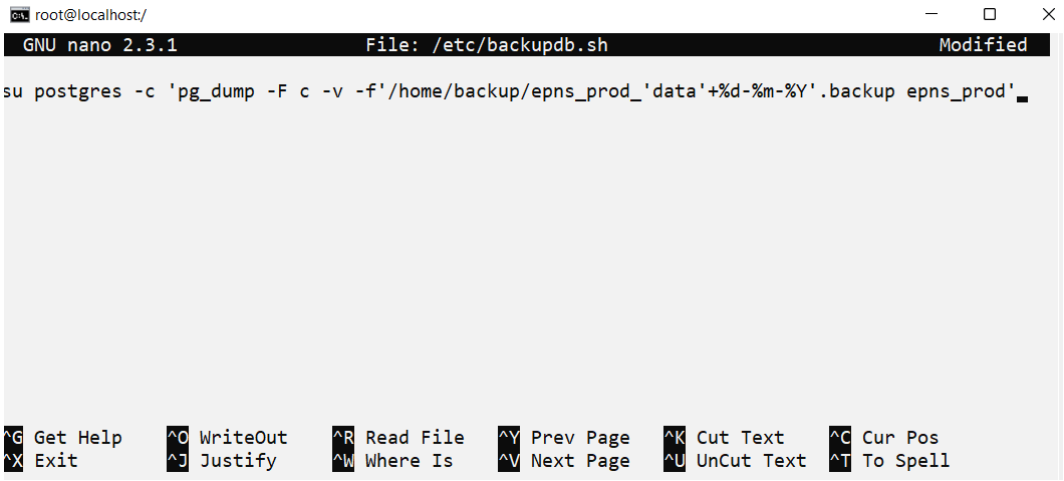


Cont..

(1) Backup & Recovery HDD Eksternal

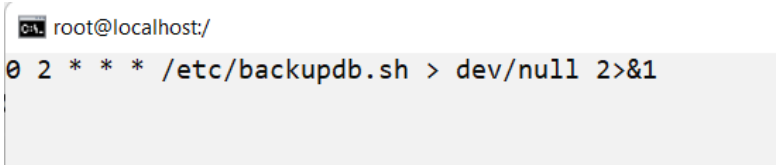
5. Automatic Backup database

Create bash shell and save with extension .sh



And do the -> crontab -e

0 2 0 * * [backup_command]



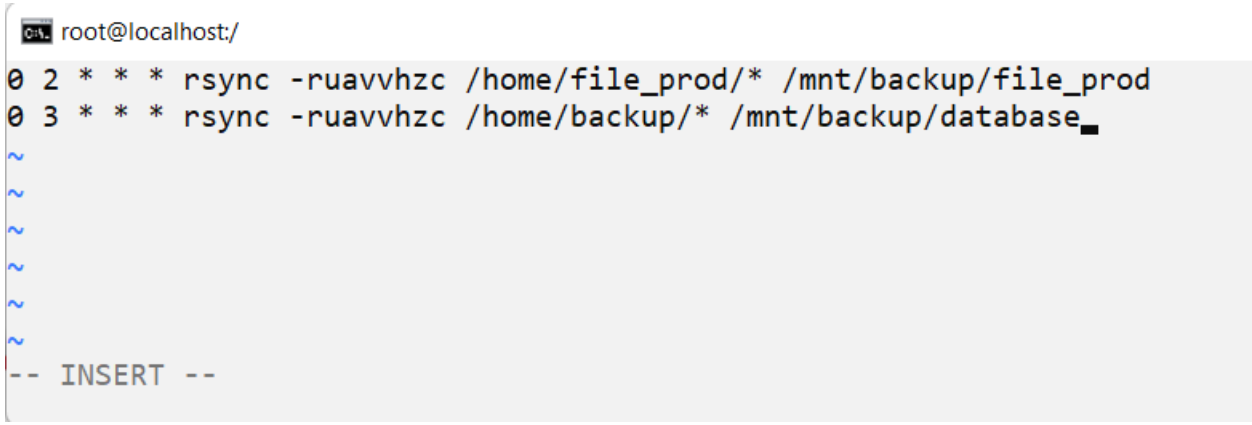
(1) Backup & Recovery HDD Eksternal

6. Synchronize file backup (DB and file_prod) from server to HDD external

And do the -> crontab -e

0 2 * * * rsync –ruavvhzc /home/file_prod/* /mnt/backup/file_prod

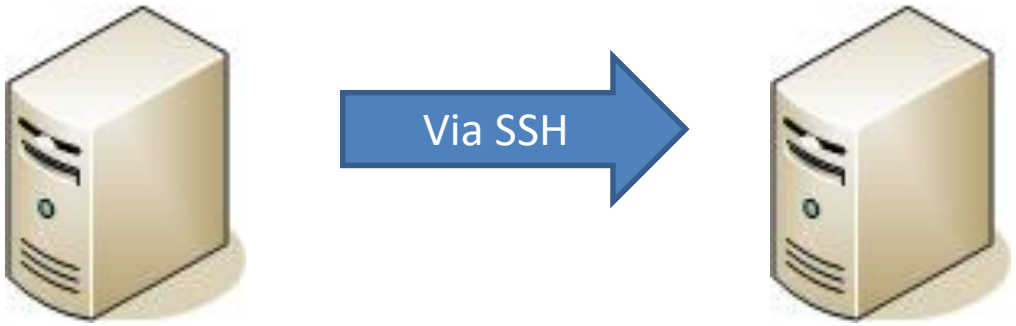
0 3 * * * rsync –ruavvhzc /home/backup/* /mnt/backup/database/



(2) Backup & Recovery Server via SSH

192.168.1.1/24
(Server Production)

192.168.1.2/24
(Server Backup)



(2) Backup & Recovery Server via SSH

Setup SSH without password.

First, we need to install:

ssh-keygen

```
root@localhost:~
[root@localhost ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
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:mj7jhlFHLoG4092oqmvhc50Nnl+aV05CUZpcwcPVM4o root@localhost.localdomain
The key's randomart image is:
+---[RSA 2048]----+
|    . .    ++o..  |
|   . . . o.++   + |
|    o . * +. o . o|
|   o . = +. E .   |
|    . o oS        |
| .  o. o . o      |
|. ..ooB  .=       |
| +...*+.+. .      |
|+oo  o+*.         |
+----[SHA256]-----+
[root@localhost ~]#
```

(2) Backup & Recovery Server via SSH

Generate the SSH copy ID:

ssh-copy-id root@ip add

Command backup db dan file upload otomatis menggunakan crontab

(2) Backup & Recovery Server via SSH

Generate the SSH copy ID:

ssh-copy-id root@[ip add server]

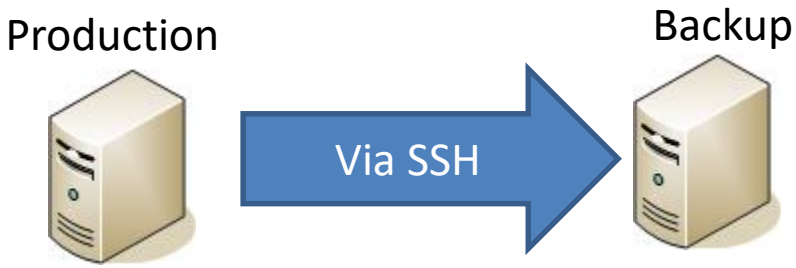
```
root@localhost:~
[root@localhost ~]# ssh-copy-id root@172.16.152.22
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.16.152.22 (172.16.152.22)' can't be established.
ECDSA key fingerprint is SHA256:wXOK0Vmps7bkNSOXKTmQEWsXIrfNqIrQhMjYsMgl+Mo.
ECDSA key fingerprint is MD5:23:47:51:d8:38:74:ef:67:74:e0:e0:e9:73:90:09:e7.
Are you sure you want to continue connecting (yes/no)?
```

(2) Backup & Recovery Server via SSH

Rsync the database backup from Server Production to Server Backup via SSH

```
#/bin/bash

###### Dump Database #######
su postgres -c "pg_dump -vFc -f /backupdb/epns_prod_`date +%d%m%Y`.backup
epns_prod" > /dev/null 2>&1

###### Delete 30 days(3 months) old file dump ##############
find /backupdb -maxdepth 1 -type f -name \*.backup.gz -mtime +30 -exec rm '{}' +

###### compress and sent to backup server db ########
gzip /backupdb/*.backup

rsync -ruvvzcah /backupdb/*.backup.gz user@ipserverbackup:/backupdb/
```

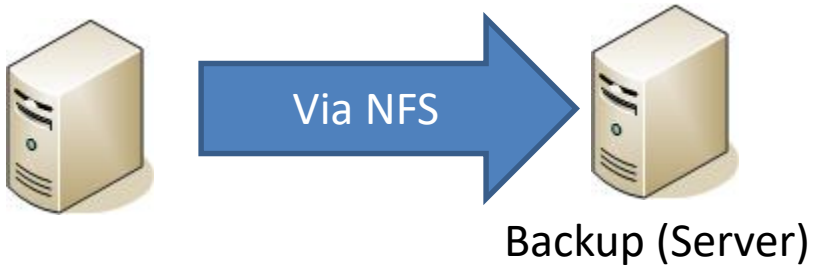
(2) Backup & Recovery Server via SSH



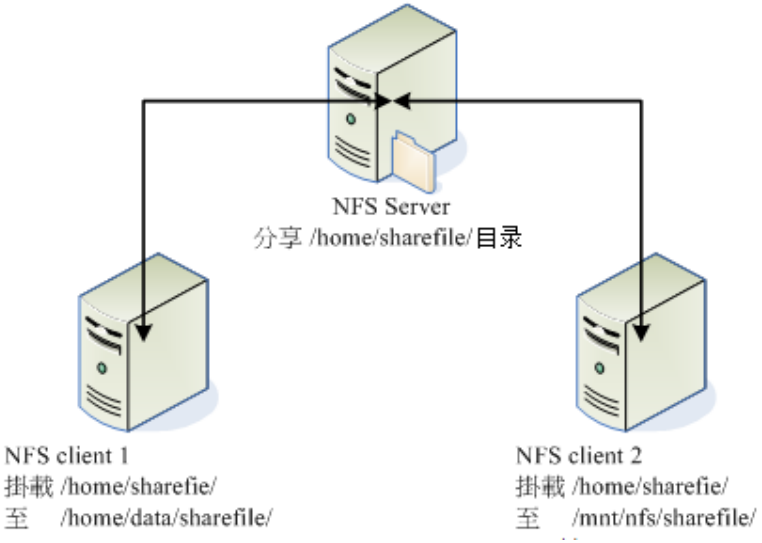
Rsync the data from Server Production to Server Backup via SSH

```
#!/bin/bash

##Rsync Data File##
rsync -ruvvzcah --rsh 'ssh -p23722' /home/file/* root@ipserverbackup:/home/file/file_prod

##Rsync Data Aplikasi##
rsync -ruvvzcah --rsh 'ssh -p23722' /home/appserv/* root@ipserverbackup:/home/appserv/

##Rsync Data Log##
rsync -ruvvzcah --rsh 'ssh -p23722' /var/log/* root@ipserverbackup:/home/backuplog/
```

Backup & Recovery Server via NFS



Network File Sharing (NFS) is **a protocol that allows you to share directories and files with other Linux clients over a network**. As like as Samba on Windows.



Backup & Recovery Server via NFS

Backup (Server)

1. Install Network File Sharing (NFS) on SERVER BACKUP

```
# sudo apt update
# sudo apt install nfs-kernel-server
```

2. Create an NFS Export Directory

```
# sudo mkdir -p /mnt/nfs_share
# sudo chown -R nobody:nogroup /mnt/nfs_share/
# sudo chmod 777 /mnt/nfs_share/
```

Backup & Recovery Server via NFS

Backup (Server)

3. Grant NFS Share Access to Client Systems

   # sudo vim /etc/exports

   # /mnt/nfs_share 192.168.43.0/24(rw,sync,no_subtree_check)

   Explanation about the options used in the above command.

   rw: Stands for Read/Write.
   sync: Requires changes to be written to the disk before they are applied.
   No_subtree_check: Eliminates subtree checking.

Backup & Recovery Server via NFS

Backup (Server)

4. Export the NFS Share Directory

```
# sudo exportfs -a
# sudo systemctl restart nfs-kernel-server
```

5. Allow NFS Access through the Firewall

```
# sudo ufw allow from 192.168.43.0/24 to any port nfs
# sudo ufw enable
# sudo ufw status
```

Backup & Recovery Server via NFS

Production (Client)

1. Install the NFS-Common Package

   # sudo apt install nfs-common

2. Create an NFS Mount Point on Client

   # sudo mkdir -p /mnt/nfs_clientshare

3. Mount NFS Server on Client System

   # sudo mount 192.168.43.234:/mnt/nfs_share /mnt/nfs_clientshare

Backup & Recovery Server via NFS

Production (Client)

4. Testing the NFS Share on Client System

cd /mnt/nfs_share/
touch file1.txt file2.txt file3.txt

ls -l /mnt/nfs_clientshare/

```
tecmint@client-server:~$
tecmint@client-server:~$ ls -l  /mnt/nfs_clientshare/
total 0
-rw-rw-r-- 1 1002 1002 0 Mar  8 20:36 file1.txt
-rw-rw-r-- 1 1002 1002 0 Mar  8 20:36 file2.txt
-rw-rw-r-- 1 1002 1002 0 Mar  8 20:36 file3.txt
tecmint@client-server:~$
tecmint@client-server:~$
```

Command backup db dan file upload otomatis menggunakan crontab

Backup & Recovery Server via NFS

Create server

#apt-get install nfs-common portmap nfs-kernel-server

Config /etc/exports

-- command share yang dibuat.

contoh : /home/ IP_CLIENT (rw, no_root_squash)

#exportfs –a

#/etc/init.d/nfs-common restart

Create client

#apt-get install nfs-common

Mounting

Config /etc/fstab

#mount –t nfs IP_SERVER_NFS:/home /mnt

Command backup db dan file upload otomatis menggunakan crontab

https://www.tecmint.com/install-nfs-server-on-ubuntu

Backup Database

File Backup database pada server Production minimal 1 minggu, sedangkan pada media Backup (server, storage, harddisk eksternal, NFS) minimal 2 bulan

Script remove file backup database :
(remove otomatis file backup database 7 hari kebelakang, pada server production)
find (lokasi backup database) -maxdepth 1 -type f -name *.backup.gz -mtime +7 -exec rm '{}' +

(remove otomatis file backup database 60 hari kebelakang, pada media backup)
find (lokasi backup database) -maxdepth 1 -type f -name *.backup.gz -mtime +60 -exec rm '{}' +

Kompresi file backup database
gzip (lokasi backup database)/*.backup

Restore Database from Backup File

Create database

```
# su postgres
$ createdb –O epns namadb
```

Ekstrak file gzip

```
# gunzip (lokasi backup db)/epns_prod_`date +%d%m%Y`.backup.gz
```

Restore Database

```
# su postgres
$ pg_restore -Fc (lokasi backup db)/epns_prod_`date +%d%m%Y`.backup -d
namadb
```

Backup Log

Backup log pada server Production minimal 2 bulan, sedangkan pada media Backup (server, storage, harddisk eksternal, NFS) minimal 4 bulan

Log pada server yang dilakukan backup, diantaranya :

- Log database
- Log web server
- Log Aplikasi SPSE
- Log System

Backup Log

Crontab info

```
🎯 A crontab file has five fields for specifying:

*   *   *   *   *   command to be executed
-   -   -   -   -
|   |   |   |   |
|   |   |   |   +--------------- **DAY OF WEEK** (0-6) (Sunday=0)
|   |   |   +------------ **MONTH** (1-12)
|   |   +--------- **DAY OF MONTH** (1-31)
|   +------ **HOUR** (0-23)
+--- **MINUTE** (0-59)
```

Backup Log

Crontab info

https://crontab.guru/

Terima Kasih