

네트워크 구성

	www	db01	storage	ns01
NAT Network	10.0.2.11/24			10.0.2.31/24
Net1 Host-only Network	192.168.56.11/24	192.168.56.21/24	192.168.56.41/24	192.168.56.31/24
Net2 Host-only Network	192.168.60.11/24	192.168.60.21/24	192.168.60.41/24	

세부 구현

단계	각 구현단계
1단계	<p>www : 웹 서버</p> <ul style="list-style-type: none"> - WordPress를 이용한 웹 서비스 제공 - DocumentRoot: /var/www/html/ - Web Server와 DB 서버는 별도로 구성함. - 웹 서비스(:80)으로 WordPress 제공 <p>ns01 : DNS 서버</p> <ul style="list-style-type: none"> - 관리 도메인 : example.com - 위에 제시된 host의 이름을 기준으로 도메인 구성 - 도메인 주소로 각 호스트에 접근 가능하도록 구성 <p>db01 DB 서버</p> <ul style="list-style-type: none"> - 외부 네트워크와의 연결이 되지 않음 - DBMS : MariaDB - 내부 네트워크에서만 접근 가능 <p>storage : Storage 서버(NFS, iSCSI 서비스 제공)</p> <ul style="list-style-type: none"> - 외부 네트워크와의 연결이 되지 않음. - 스토리지 서비스는 스토리지 네트워크를 통해 제공함. - 호스트에 하드디스크(10G) 2개 추가하여 - 첫번째 디스크를 사용하여 NFS 서비스 구성 - 두번째 디스크를 사용하여 iSCSI 서비스 구성

2단계	<p>www : 웹 서버 Web Site 1 : WordPress - 웹 사이트 주소 : http://www.example.com/wordpress - WordPress를 이용한 웹 서비스 제공 - DocumentRoot : /var/www/html/ - Web Server와 DB 서버는 별도로 구성함. - 웹 서비스(:80)으로 WordPress 제공</p> <p>Web Site 2 : 단순 웹 사이트(web01) - 웹 사이트 주소 : http://web01.example.com:8080/ - index.html 파일에 다음 내용 포함. "Welcome to Apache Webserver (web01)" - DocumentRoot : /srv/www/web01 (NFS 제공 경로) - Port: 8080</p> <p>www02 : 웹 서버 Web Site 2 : 단순 웹 사이트(web02) - 웹 사이트 주소 : http://web02.example.com:8080/ - index.html 파일에 다음 내용 포함. "Welcome to Apache Webserver (web02)" - DocumentRoot : /srv/www/web02 (NFS 제공 경로) - Port: 8080</p> <p>ns01 : DNS 서버 - 관리 도메인 : example.com - 위에 제시된 host의 이름을 기준으로 도메인 구성 - 도메인 주소로 각 호스트에 접근 가능하도록 구성 - www, www02 호스트의 Web Site 2에 대해 DNS 기반 로드밸런싱을 구성. http://internal-portal.example.com:8080/ 접속시 "Welcome to Apache Webserver (web0X)" 로 접속됨. (web01, web02 번갈아 가며 접속됨.)</p> <p>db01 DB 서버 - 외부 네트워크와의 연결이 되지 않음. - DBMS : MariaDB - 내부 네트워크에서만 접근 가능. - db01의 mariadb DBMS의 데이터는 storage 서버에서 iSCSI로 제공된 디스크에 저장되도록 구성함. iSCSI 디스크를 연결하고 디스크 전체 공간을 사용하도록 파티셔닝 한 후 ext4 파일시스템으로 초기화 하여 /datastore 로 영구마운트 설정.</p> <p>storage : Storage 서버(NFS, iSCSI 서비스 제공) - 외부 네트워크와의 연결이 되지 않음. - 스토리지 서비스는 스토리지 네트워크를 통해 제공. - 호스트에 하드디스크(10G) 2개 추가 - 첫번째 디스크를 사용하여 NFS 서비스 구성 (NFS는 Net2를 통해서 제공) - 두번째 디스크를 사용하여 iSCSI 서비스 구성 (iSCSI는 Net2를 통해서 제공)</p>
-----	---

리눅스 프로젝트

1단계

프로젝트를 위해서 네 개의 가상 머신이 필요합니다.

그렇기에 각 가상 머신 별로 아래와 같은 네트워크로 설정 했습니다.

	www	db01	storage	ns01
NAT Network	10.0.2.11/24	-	-	10.0.2.31/24
Net1 Host-only Network	192.168.56.11/24	192.168.56.21/24	192.168.56.41/24	192.168.56.31/24
Net2 Host-only Network	192.168.60.11/24	192.168.60.21/24	192.168.60.41/24	-

1번 가상머신 부터 4번 가상 머신 까지 www, db01, storage, ns01로 설정했으며

NAT Network는 enp0s3, Net1 Host-only Network 는 enp0s8,

Net2 Host-only Network 는 enp0s9으로 설정 했습니다.

각 서버에서 사용하지 않는 파트의 NIC는 Disable 처리 했습니다.

순서대로 hostname을 www01,db01,storage01,dns01로 설정하였습니다.

```
nmcli connection add con-name enp0s9 type ethernet ifname enp0s9
nmcli connection show
nmcli connection modify enp0s9 ipv4.addresses 192.168.56.11/24
nmcli connection modify enp0s9 ipv4.method manual
nmcli connection up enp0s9
```

```
[root@www01 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.11 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe85:4141 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:85:41:41 txqueuelen 1000 (Ethernet)
    RX packets 4523 bytes 4205067 (4.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3410 bytes 1053309 (1.0 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.11 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fe50:29f5 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:50:29:f5 txqueuelen 1000 (Ethernet)
    RX packets 753 bytes 194692 (190.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 639 bytes 107439 (104.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s9: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.60.11 netmask 255.255.255.0 broadcast 192.168.60.255
    inet6 fe80::b313:fb2a:ele9:6a9b prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:42:6f:2c txqueuelen 1000 (Ethernet)
    RX packets 9163 bytes 3257274 (3.1 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
```

```
[root@dns01 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.31 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fec0:581f prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:c0:58:1f txqueuelen 1000 (Ethernet)
    RX packets 3973 bytes 2650392 (2.5 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3352 bytes 425804 (415.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.31 netmask 255.255.255.0 broadcast 192.168.56.
255
    inet6 fe80::a00:27ff:fe87:3d3f prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:87:3d:3f txqueuelen 1000 (Ethernet)
    RX packets 60 bytes 8040 (7.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 44 bytes 4761 (4.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s9: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 08:00:27:02:98:f4 txqueuelen 1000 (Ethernet)
    RX packets 7 bytes 420 (420.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
```

```
[root@db01 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 08:00:27:e8:17:b2 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.21 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fe9c:e9fc prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:fc:e9:fc txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 41 bytes 4578 (4.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s9: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.60.21 netmask 255.255.255.0 broadcast 192.168.60.255
    inet6 fe80::992:1aa1:63e5:7af8 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:d1:96:48 txqueuelen 1000 (Ethernet)
    RX packets 17791 bytes 25124880 (23.9 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2721 bytes 462345 (451.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@storage01 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 08:00:27:13:d8:6f txqueuelen 1000 (Ethernet)
    RX packets 468 bytes 55444 (54.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 303 bytes 28611 (27.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.41 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fe06:6273 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:06:62:73 txqueuelen 1000 (Ethernet)
    RX packets 324 bytes 40657 (39.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 57 bytes 5918 (5.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s9: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.60.41 netmask 255.255.255.0 broadcast 192.168.60.255
    inet6 fe80::a373:c59c:d917:edf6 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:0b:ac:d9 txqueuelen 1000 (Ethernet)
    RX packets 31610 bytes 7804067 (7.4 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 67527 bytes 68788744 (65.6 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

웹 서버로 사용하는 www의 가상 머신의 서버의 조건은 다음과 같습니다.

- WordPress를 이용한 웹 서비스 제공
 - DocumentRoot 56.: /var/www/html/
 - Web Server와 DB 서버는 별도로 구성함.
 - 웹 서비스(:80)으로 WordPress 제공
- wordpress구성에 사용한 주요 코드

wordpress구성에 사용한 주요 코드

```
yum install httpd
vim /etc/http/conf.d/01-www.example.com.conf
<VirtualHost *:80>
ServerAdmin webmaster@example.com
DocumentRoot "/var/www/html"
ServerName www.example.com
ErrorLog "/var/log/httpd/www.example.com-error_log"
CustomLog "/var/log/httpd/www.xample.com-access_log" common
</VirtualHost>
<Directory "/var/www/html">
Options Indexes FollowSymLinks
AllowOverride None
Require all granted
</Directory>
yum install mariadb-server mariadb
yum install php
yum install php php-mysqlnd
systemctl start httpd.service
firewall-cmd --add-service=http
firewall-cmd --add-service=http --permanent
cd /var/www/html
vim index.html
<a href="/wordpress">Go to WordPress Page</a>
wget https://wordpress.org/latest.zip
unzip latest.zip
vim wp-config.php
```

DNS 서버로 사용하는 ns01의 가상 머신의 서버의 조건은 다음과 같습니다.

- 관리 도메인 : example.com
- 위에 제시된 host의 이름을 기준으로 도메인 구성
- 도메인 주소로 각 호스트에 접근 가능하도록 구성DNS 서버 구성 주요 코드

```
yum install bind
yum install nfs-utils
vim /etc/named.conf
```

```

yum start named.service
yum enable named.service
firewall-cmd --add-service=dns
firewall-cmd --add-service=dns --permanent
vim /etc/named.rfc1912.zones
#domain zones
zone "example.com" IN {
    type master;
    file "example.com.zone";
    allow-update { none; };
};
vim /var/named/example.com.zone
chgrp named /var/named/example.com.zone
vim /etc/resolv.conf

```

```

vim /var/named/example.com.zone
$TTL 1D
@      IN SOA  ns hostmaster (
1      ; serial
1D     ; refresh
1H     ; retry
1W     ; expire
3H )   ; minimum
@      IN NS   ns
      IN A    192.168.56.31
ns     IN A    192.168.56.31
www    IN A    192.168.56.11
db     IN A    192.168.56.21
storage IN A    192.168.56.41

```

DB 서버로 사용하는 db01의 가상 머신의 서버의 조건은 다음과 같습니다.

- 외부 네트워크와의 연결이 되지 않음.
- DBMS : MariaDB

- 내부 네트워크에서만 접근 가능.

DB서버 구성 주요 코드

```
yum install mariadb-server mariadb
systemctl start mariadb.service
systemctl enable mariadb.service
firewall-cmd --add-service=mysql
firewall-cmd --add-service=mysql --permanent
mysql -h localhost -u root
CREATE DATABASE nux1;
CREATE USER 'db03'@'192.168.56.21' IDENTIFIED BY '1234'
GRANT ALL PRIVILEGES ON nux1.* TO 'db03'@'192.168.56.%';
FLUSH PRIVILEGES;
```

Storage 서버로 사용하는 storage의 가상 머신의 서버의 조건은 다음과 같습니다.

- 외부 네트워크와의 연결이 되지 않음.
- 스토리지 서비스는 스토리지 네트워크를 통해 제공.
- 호스트에 하드디스크(10G) 2개 추가
- 추가한 첫번째 디스크를 사용하여 NFS 서비스 구성
- 추가한 두번째 디스크를 사용하여 iSCSI 서비스 구성

STORAGE서버 구성 주요 코드

```
yum install nfs-utils
yum install targetcli
systemctl start nfs-server.service
systemctl enable nfs-server.service
firewall-cmd --add-service=nfs
firewall-cmd --add-service=nfs --permanent
vim /etc/exports
exportfs -r
fdisk /dev/sdb
mkfs -t xfs /dev/sdb1
```

```

mount -t xfs /dev/sdb1 /mnt/test
systemctl start target.service
systemctl enable target.service
firewall-cmd --add-port=3260/tcp
firewall-cmd --add-port=3260/tcp --permanent
targetcli
/backstores/block create name=storage dev=/dev/sdc
/iscsi create wwn=iqn.2024-05.com.example:www
iscsi/iqn.2024-05.com.example:www/tpg1/acls create wwn=iqn2024-
/iscsi/iqn.2024-05.com.example:www/tpg1/luns create storage_obj
/iscsi/iqn.2024-05.com.example:www/tpg1/portals delete 0.0.0.0
/iscsi/iqn.2024-05.com.example:www/tpg1/portals create 192.168.

```

2단계

워드 프레스 사이트 구축

www : 웹 서버

Web Site 1 : WordPress

- Web Server와 DB 서버는 별도로 구성함.

서버 별도 구성을 위한 코드

WWW SERVER

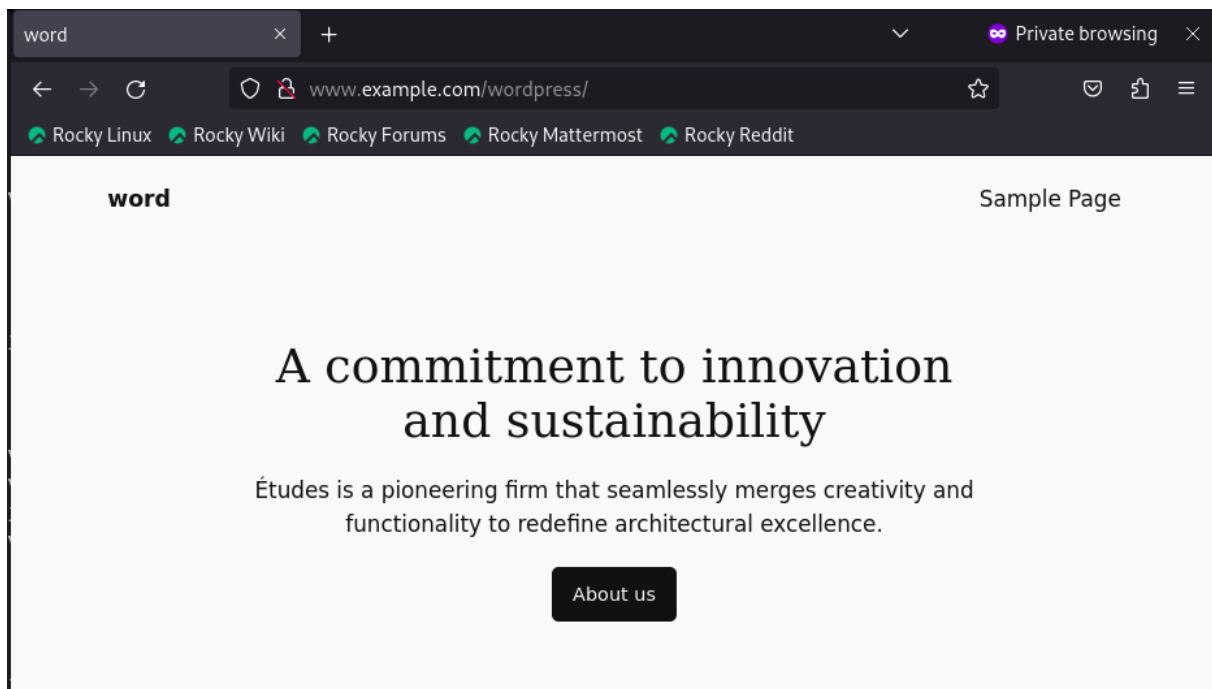
```

yum install mariadb-server mariadb
systemctl start mariadb.service
systemctl enable mariadb.service
firewall-cmd --add-service=mysql
firewall-cmd --add-service=mysql --permanent
vim /var/www/html/wordpress/wp-config.php
define( 'DB_NAME', 'mysql' );
define( 'DB_USER', 'db03' );
define( 'DB_PASSWORD', '1234' );
define( 'DB_HOST', '192.168.56.21' );
mysql -h 192.168.56.21 -u db03 -p

```

DB SERVER

```
vim /etc/my.cnf.d/mariadb-server.cnf
[mysqld]
datadir=/mnt/storage/mysql
socket=/mnt/storage/mysql/mysql.sock log-error=/var/log/mariadb
pid-file=/run/mariadb/mariadb.pid
cp -r /var/lib/mysql /mnt/storage/
chmod 777 /mnt/storage/mysql
chown -R mysql /mnt/storage/mysql
chgrp -R mysql /mnt/storage/mysql
```



서버 구축 및 도메인 로드 밸런싱 구현 여부

www, www02 호스트의 Web Site 2에 대해 DNS 기반 로드밸런싱을 구성.
(web01, web02 번갈아 가며 접속됨.)

구성을 위한 코드

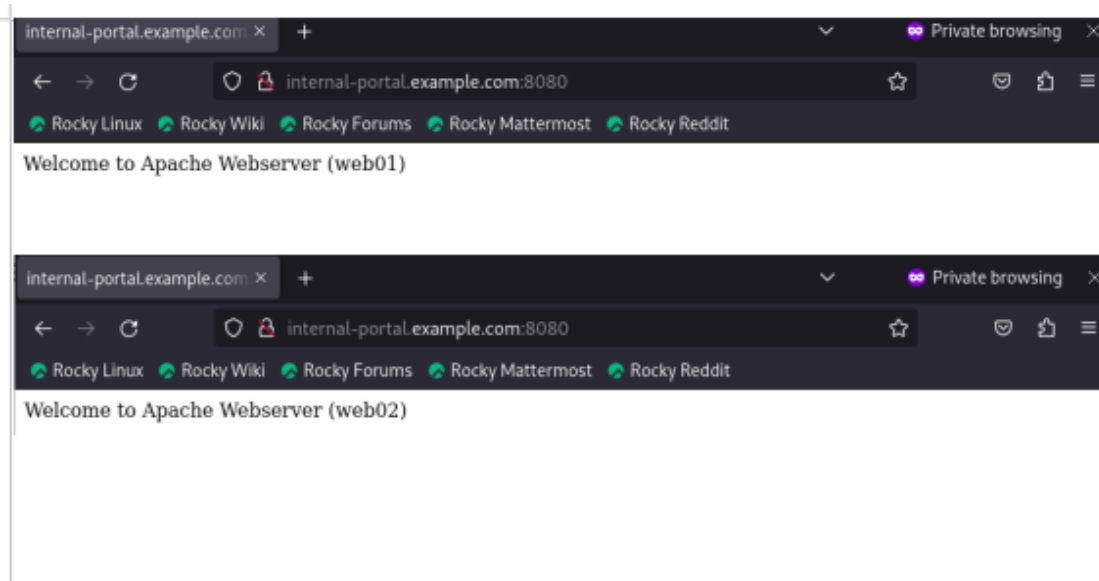
DNS SERVER

```
vim /var/named/example.com.zone
internal-portal 1 IN A 192.168.60.11
```

```
internal-portal 2 IN A 192.168.56.11
```

WWW SERVER

```
vim /etc/httpd/conf.d/00-port-8080.conf
<VirtualHost 192.168.56.11:8080>
ServerAdmin webmaster@example.com
DocumentRoot "/srv/www/web01"
ErrorLog "/srv/www/web01/web01-error_log"
CustomLog "/srv/www/web01/web01-access_log" common
</VirtualHost>
<Directory "/srv/www/web01">
Options Indexes FollowSymLinks
AllowOverride None
Require all granted
</Directory>
<VirtualHost 192.168.60.11:8080>
ServerAdmin webmaster@example.com
DocumentRoot "/srv/www/web02"
ErrorLog "/srv/www/web01/web02-error_log"
CustomLog "/srv/www/web01/web02-access_log" common
</VirtualHost>
<Directory "/srv/www/web02">
Options Indexes FollowSymLinks
AllowOverride None
Require all granted
</Directory>
```



iSCSI와 DB 연동 여부

db01 DB 서버

- db01 의 mariadb DBMS의 데이터는 storage 서버에서 iSCSI로 제공된 디스크에 저장 되도록 구성.

구성을 위한 코드

DB SERVER

```
yum install iscsi-initiator-utils
vim /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.2024-05.com.example:host1
systemctl start iscsi.service
systemctl enable iscsi.service
iscsiadm -m discovery -t sendtargets -p 192.168.60.41:3260
iscsiadm -m node -T iqn.2024-05.com.example:servera -p 192.168.
iscsiadm -m session -P 3
fdisk /dev/sdb
mkfs -t ext4 /dev/sdb1
mount -t ext4 /dev/sdb1 /mnt/storage
mkdir /mnt/storage/mysql
vim /etc/my.cnf.d/mariadb-server.cnf
[mysqld]
datadir=/mnt/storage/mysql
```

```
socket=/mnt/storage/mysql/mysql.sock log-error=/var/log/mariadb/
pid-file=/run/mariadb/mariadb.pid
cp -r /var/lib/mysql /mnt/storage/
chmod 777 /mnt/storage/mysql
chown -R mysql /mnt/storage/mysql
chgrp -R mysql /mnt/storage/mysql
systemctl start mariadb.service
```