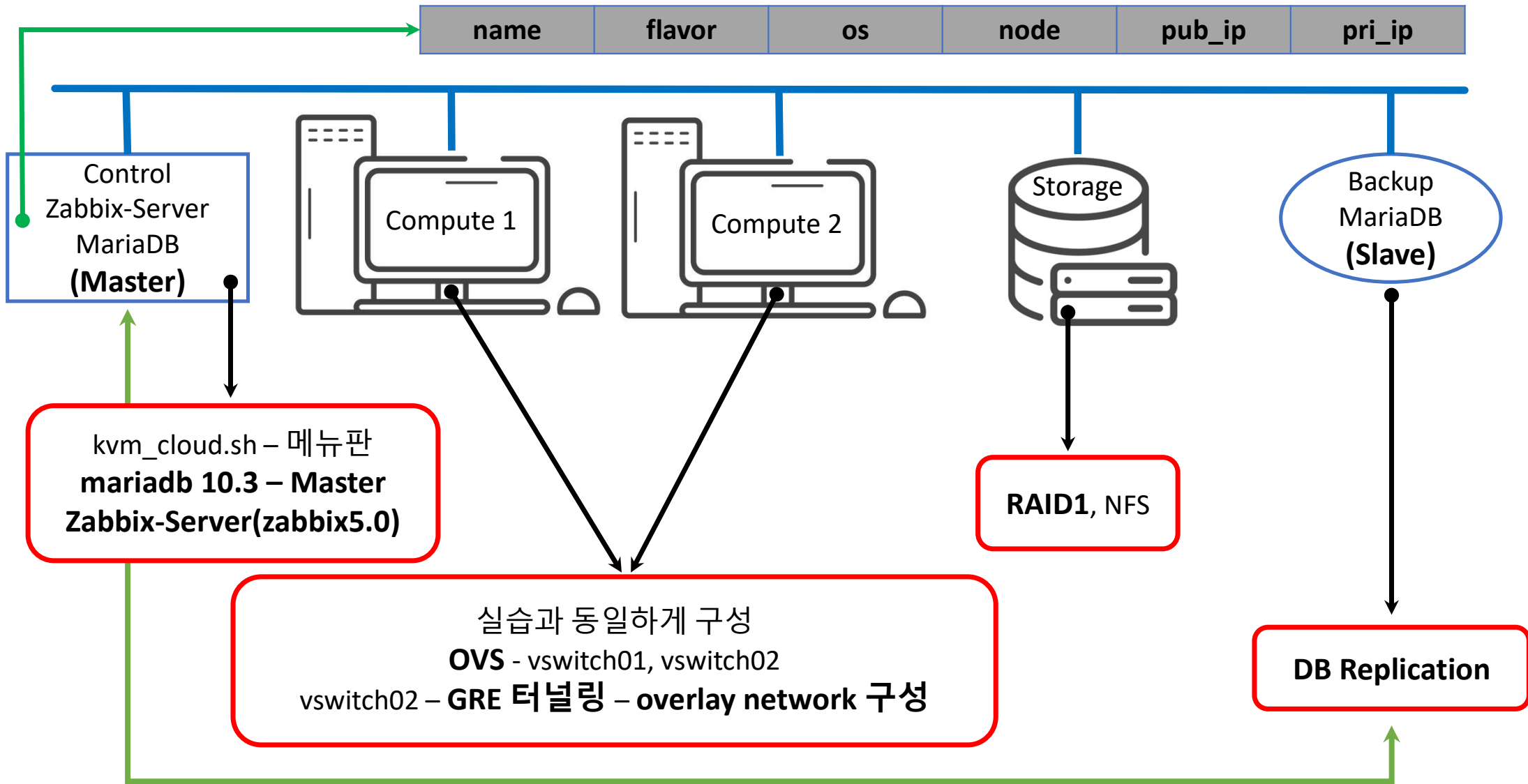


Kakao cloud

1.4 - 1.8 프로젝트

카카오 클라우드 엔지니어 4기
김정우, 김희원, 백승헌, 유태균, 윤순상


시나리오



MENU

KAKAO Cloud Engineer KVM Project

1. 가상 인스턴스 생성하기
2. 가상 인스턴스 목록 확인하기
3. 가상 인스턴스 삭제하기
0. 종료하기

메뉴번호를 선택하세요! : 

- [메뉴판 쉘 스크립트 github](#)

- git clone https://github.com/Yunsoonsang/my_shell.git

CODE <변수와 메인>

Control

Compute

Storage

MariaDB

Zabbix

```
# main
while [ 1 ]
do
    menu
    case $option in
    0)
        echo -e "\t2초 뒤 프로그램을 종료합니다!"
        break;;
    1)
        create_instance;;
    2)
        show_instance;;
    3)
        delete_instance;;
    *)
        clear
        echo -e "\t존재하지 않는 메뉴 번호를 선택했습니다. 다시 메뉴를 선택해주세요.;;"
    esac
    echo -en "\n\n\t\t아무 키나 입력하시면 다시 진행됩니다."
    read -n 1 line
done
sleep 2
clear
```

```
volume="" # 이미지 볼륨
name="" # 인스턴스 이름이자 hostname
flavor="" # 인스턴스 스펙 = cpu, ram
cpu="" # cpu개수
ram="" # ram용량
com="" # VM을 생성할 compute의 이름
vmcount="" # 생성할 인스턴스의 개수
com1_usage="" # compute1 cpu 사용량
com2_usage="" # compute2 cpu 사용량
```

CODE <인스턴스 생성>

```
# 메뉴1 - 가상 인스턴스 생성하기
function create_instance {
    echo -e "\n\t가상 인스턴스 생성하기 메뉴 실행"
    # 인스턴스 이미지 볼륨 선택하기
    menu_volume
    # 인스턴스 이름 입력받기 - hostname으로도 사용
    menu_name
    # flavor 선택메뉴
    menu_flavor
    # 생성할 인스턴스의 개수 입력받기
    menu_vmcount
    #echo -e "\t\t현재까지 입력된 데이터 출력"
    #echo -e "\tvolume = $volume"
    #echo -e "\tname = $name"
    #echo -e "\tflavor = $flavor"
    #echo -e "\tvmcount = $vmcount"
    echo -e "\t인스턴스를 생성할 노드를 결정하는 중 입니다. 잠시 기다려 주세요..."
    com1_usage=$(ssh compute1 'cpu.sh')
    com2_usage=$(ssh compute2 'cpu.sh')
    echo -e "\tcom1:$com1_usage, com2:$com2_usage"
    if [ $com2_usage -eq $com1_usage ]
    then
```

Control

Compute

Storage

MariaDB

Zabbix

CODE <인스턴스 생성>

```
then
    # 두 노드의 cpu사용량이 같으므로 default로 compute1에 설치
    com="compute1"
    create compute1
elif [ $com2_usage -gt $com1_usage ]
then
    # compute1에 인스턴스 생성
    com="compute1"
    create compute1
else
    # compute2에 인스턴스 생성
    com="compute2"
    create compute2
fi
```

Control

Compute

Storage

MariaDB

Zabbix

CODE <내부 설정>

```
# 인스턴스 이름 입력받기
function menu_name {
    while [ 1 ]
    do
        clear
        # 인스턴스 이름 입력받기 - hostname으로도 사용
        echo -e "\t인스턴스의 이름을 입력하세요."
        echo -e "\t알파벳 소문자와 숫자로 최소 5글자 ~ 최대 10글자로 구성하세요. 첫 시작은 반드시 문자여야 합니다."
        echo -en "\t 인스턴스 이름 : "
        read name
        name=$(echo $name | gawk '/^[a-z]{1}[a-z0-9]{4,9}$/ {print $0}')
        if [ -z $name ]
        then
            clear
            echo -e "\n\t잘못된 형식입니다. 패턴:[a-z]{1}[a-z0-9]{4,9}"
            echo -en "\n\n\t\t아무 키나 입력하시고 메뉴를 선택해주세요!"
            read -n 1 line
        else
            break
        fi
    done
}
```

CODE <내부 설정>

Control

Compute

Storage

MariaDB

Zabbix

```
#!/bin/bash

cpu=$(sar 1 5 | grep Average | gawk '{print $8}')
cpu=$(echo "scale=2; 100 * (100 - ${cpu})" | bc)
cpu=$(echo $cpu | gawk -F. '{print $1}')
echo $cpu
```


CODE <인스턴스 생성 함수>

Control

Compute

Storage

MariaDB

Zabbix

```
function create {
    if [ $1 = "compute1" ]
    then
        echo -e "\tcompute1에 생성"
        if [ $vmcount -eq 1 ]
        then
            echo -e "\t\t$com에 인스턴스 1개 생성 시작!!!"
            # 현재는 매우 간단하게 할 수 있도록 구현 더 추가해야함
            ssh compute1 "cp /shared/${volume} /shared/${name}.qcow2"
            ssh compute1 "virt-customize -a /shared/${name}.qcow2 --root-password password:
test123 --hostname $name --run-command 'useradd centos' --run-command 'mkdir /home/centos/.ssh' --run
-command 'chmod 700 /home/centos/.ssh' --run-command 'chown centos:centos /home/centos/.ssh' --upload
/shared/authorized_keys:/home/centos/.ssh/authorized_keys --upload /net/ifcfg-eth1:/etc/sysconfig/ne
twork-scripts/ifcfg-eth1 --selinux-relabel"
            ssh compute1 "virt-install --name ${name} --vcpus ${cpu} --ram ${ram} --disk
/shared/${name}.qcow2 --import --network=bridge:vswitch01,model=virtio,virtualport_type=openvswitch --n
etwork=bridge:vswitch02,model=virtio,virtualport_type=openvswitch --noautoconsole"
            sleep 3
            # 생성완료를 판단하기 위해 ip생성을 기다리자 근데 지금은 추가 X
            echo -e "\t\t$com에 인스턴스 1개 생성 완료!!!"
            # 데이터베이스 업데이트 추가
            upload_database $name $flavor $volume $com
        else
        fi
    fi
}
```

CODE <DB>

```
Database changed
MariaDB [kakaodb]> select * from instance_info;
+-----+-----+-----+-----+-----+-----+
| name   | flavor | os       | node   | pub_ip | pri_ip |
+-----+-----+-----+-----+-----+-----+
| centos1 | m1.small | CentOS-7.qcow2 | compute1 | NULL | NULL |
| centos1-1 | m1.small | CentOS-7.qcow2 | compute1 | NULL | NULL |
| centos1-2 | m1.small | CentOS-7.qcow2 | compute1 | NULL | NULL |
| centos2 | m1.small | CentOS-7.qcow2 | compute1 | NULL | NULL |
| centos3 | m1.small | CentOS-8.qcow2 | compute2 | NULL | NULL |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.000 sec)

MariaDB [kakaodb]>
```

name	flavor	os	node	pub_ip	pri_ip
Primary key varchar(20)	varchar(20) NOT NULL	varchar(20) NOT NULL	varchar(20) NOT NULL	varchar(100)	varchar(100)

```
function upload_database {
    echo -e "\t데이터베이스 업데이트 진행"
    echo -e "\t저장할 데이터 name=$1, flavor=$2, os=$3, node=$4"
    mysql kakaodb -u rep -ptest123 << EOF
insert into instance_info(name, flavor, os, node) values('$1', '$2', '$3', '$4');
EOF
    echo -e "\t데이터베이스 업데이트 완료"
}
```

OVS

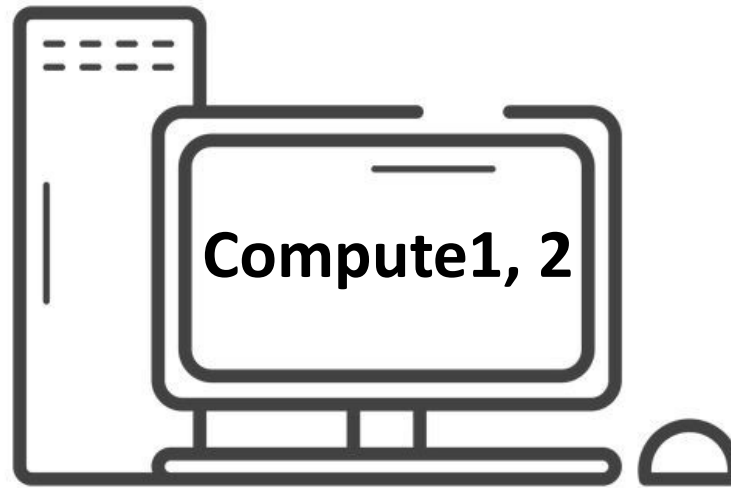
Control

Compute

Storage

MariaDB

Zabbix



강사님의 실습내용과 동일하게 ovs 세팅 하였습니다.

RAID1 & NFS server

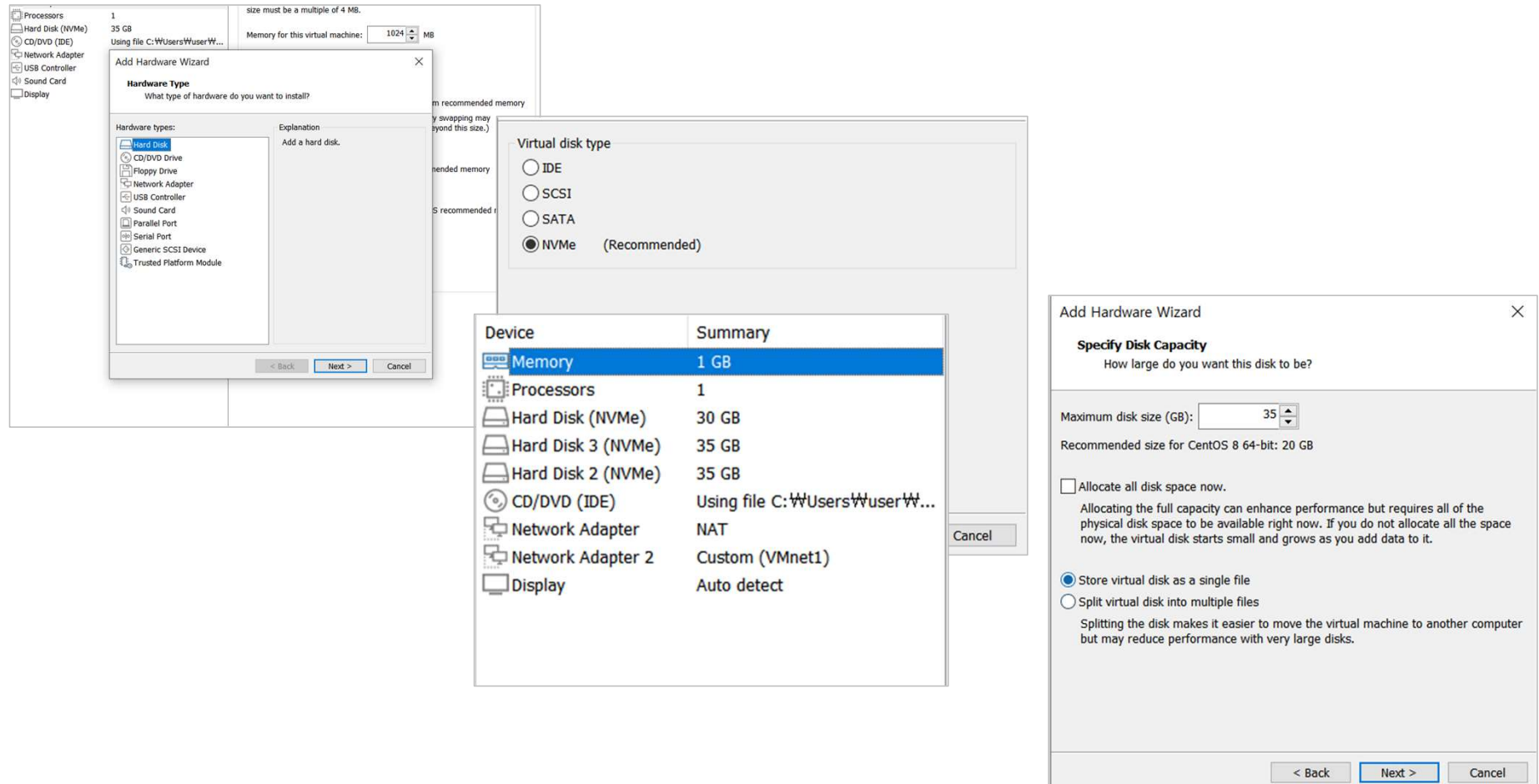
Control

Compute

Storage

MariaDB

Zabbix



RAID1 & NFS server

Control

Compute

Storage

MariaDB

Zabbix

```
Disk /dev/nvme0n2: 35 GiB, 37580963840 bytes, 73400320 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/nvme0n3: 35 GiB, 37580963840 bytes, 73400320 sectors
Units: sectors of 1 * 512 = 512 bytes
```

```
# dnf -y install mdadm nfs-utils
```

```
# fdisk -l
```

미러링 생성

```
# mdadm --create /dev/md1 --level=1 --raid-devices=2
/dev/nvme0n2 /dev/nvme0n3 (물어보는 문구 나오면 yes)
```

RAID1 & NFS server

1. mdadm --detail --scan -> 레이드 생성 확인

2. 레이드 형성 했으니 사용을 위해서는 **포맷 후 마운트**

```
# mkfs.ext4 /dev/md1 or mkfs -t ext4 /dev/md1
```

```
# mkdir /raid1; chmod 777 /raid1;
```

```
# mount /dev/md1 /raid1
```

반드시 마운트 정보 /etc/fstab에 추가

```
/dev/md1          /raid1            ext4 defaults    0 0
```

3. /etc/exports 작성

4. systemctl enable nfs-server --now

5. Compute들은 /raid1 -> /shared로 마운트 후 fstab에 마운트 정보 기록

Control

Compute

Storage

MariaDB

Zabbix

DB Replication <SLAVE>

1. Master, Slave설정을 하면 간단하게 replication이 가능하다.

2. CentOS-8의 default로 설치되는 10.3버전으로 진행했다.

3. create user 'rep'@'%' identified by 'test123';

4. grant replication slave on *.* to 'rep'@'%';
flush privileges;

5. vi /etc/my.cnf # 아래 내용 추가

[mysqld]

server_id = 2

relay_log = slave-relay-bin (로그파일 저장위치지정)

read-only (읽기전용)

systemctl restart mariadb

Control

Compute

Storage

MariaDB

Zabbix

DB Replication <MASTER>

Control

```
1. create user 'rep'@'%' identified by 'test123';  
grant replication slave on *.* to 'rep'@'%';  
flush privileges;
```

Compute

```
2. GRANT ALL PRIVILEGES ON *.* TO 'rep'@'localhost' IDENTIFIED  
BY 'test123' WITH GRANT OPTION;  
FLUSH PRIVILEGES;
```

Storage

MariaDB

```
3. vi /etc/my.cnf # 아래 내용 추가  
[mysqld]
```

Zabbix

```
server_id = 1  
log_bin = mysql-bin #파일위치지정  
max_binlog_size = 100M #파일 사이즈 지정  
expire_logs_days = 7 #log파일 포맷주기 지정  
systemctl restart mariadb  
show master status;
```


DB Replication <MASTER - Cnotrol>

Control

show master status;

Compute

Storage

MariaDB

Zabbix

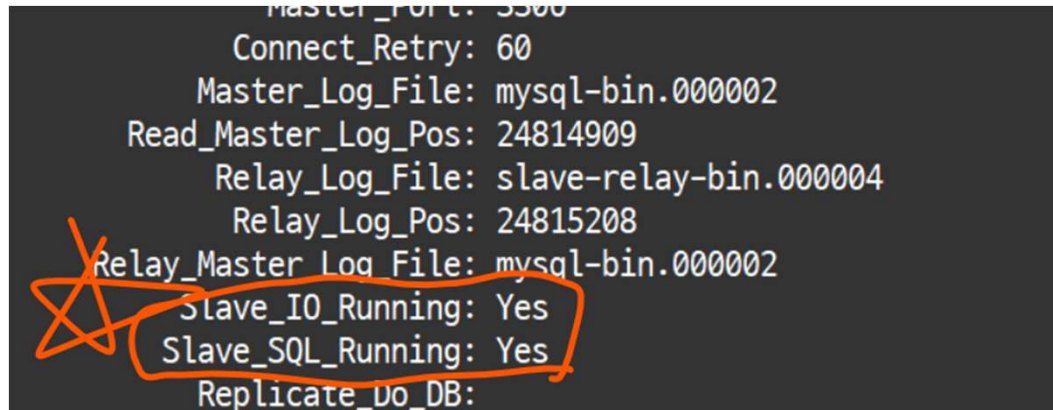
```
MariaDB [(none)]> show master status;
```

File	Position	Binlog_Do_DB	Binlog_Ignore_DB
mysql-bin.000002	24746797		

```
1 row in set (0.001 sec)
```

```
MariaDB [(none)]> █
```

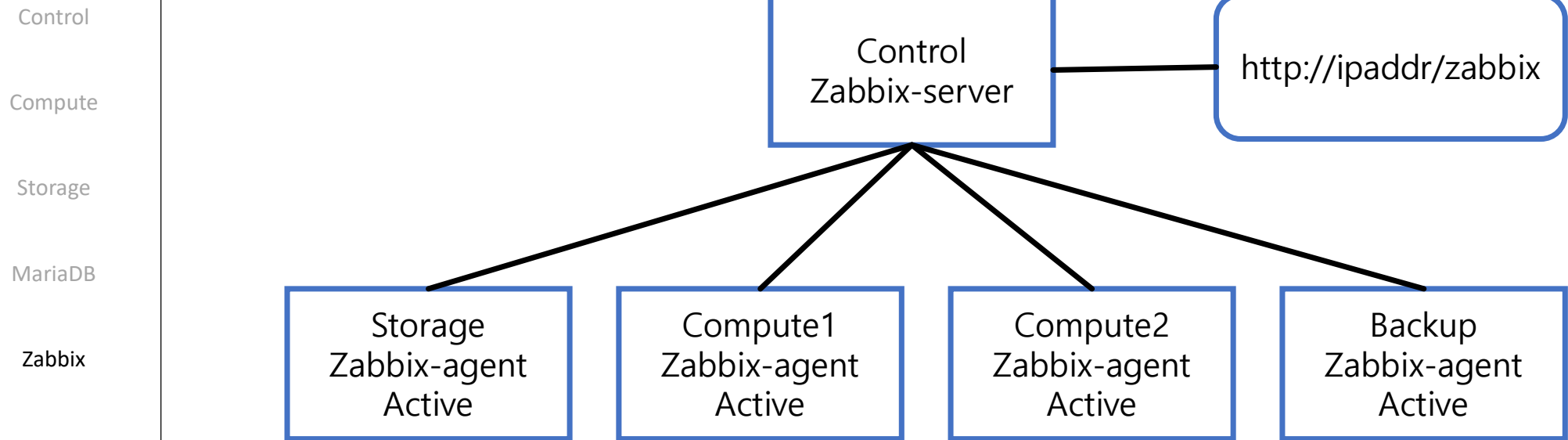
DB Replication <SLAVE - Cnontrol>



```
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000002
Read_Master_Log_Pos: 24814909
Relay_Log_File: slave-relay-bin.000004
Relay_Log_Pos: 24815208
Relay_Master_Log_File: mysql-bin.000002
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Replicate_Do_DB:
```

1. `mysql -u root -p`
2. change master to
`master_host='192.168.1.99', #control 서버 ip`
3. `master_user='rep',` #DB 접속 id
4. `master_password='test123',` #DB 접속 pw
5. `master_port=3306,` #DB 접속 port
`master_log_file='mysql-bin.000002',`
`master_log_pos= 24746797;`
6. `start slave;`

ZABBIX



CentOS-8 Zabbix 5.x

INSTALL

1. `dnf install -y httpd mariadb mariadb-devel mariadb-server`
나머지는 이미 있을 것. mariadb-devel은 추가 설치

2. `rpm -ivh`
https://repo.zabbix.com/zabbix/5.0/rhel/8/x86_64/zabbix-release-5.0-1.el8.noarch.rpm

3. `dnf clean all`
`dnf -y install zabbix-server-mysql zabbix-web-mysql zabbix-apache-conf zabbix-agent`

4. `mysql -u root -p`로 로그인
`create database zabbix character set utf8 collate utf8_bin;`

5. `grant all privileges on zabbix.* to zabbixid@localhost identified by 'password';`

6. `Flush privileges;`

Control

Compute

Storage

MariaDB

Zabbix

INSTALL

Control

Compute

Storage

MariaDB

Zabbix

```
Database changed
MariaDB [zabbix]> show tables;
+-----+
| Tables_in_zabbix |
+-----+
| acknowledges     |
| actions           |
| alerts            |
| application_discovery |
| application_prototype |
+-----+
```

1. `zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql zabbix -u zabbixid -ppassword`

2. 이후 다시 mysql로 들어가서 zabbix데이터베이스의 table을 보면

3. vi /etc/zabbix/zabbix_server.conf

4. 91줄 주석제거, 100줄, 116줄, 124줄 Dbname, userid, password 알맞게 기입

INSTALL

Control

1. vi /etc/php-fpm.d/zabbix.conf

마지막 줄 세미콜론(;)삭제 후 Asia/Seoul로 변경

Compute

2. systemctl enable zabbix-server ; systemctl enable zabbix-agent ;

Storage

3. systemctl enable httpd ; reboot

MariaDB

4. http://control_ipaddr/zabbix로 접속하여
알맞은 DB_name, User, password를 기입
Next step를 계속해 넘어가면 된다.

Zabbix

5. Zabbix 로그인

ID : Admin

PW : zabbix (default)

WEB

Control

Compute

Storage

MariaDB

Zabbix

22

ZABBIX << >> Global view

전체 대시보드 / Global view

시스템 정보

파라미터	값	자세히
Zabbix 서버 가동 중	예	localhost:10051
Number of hosts (enabled/disabled)	5	5 / 0
Number of templates	218	
아이템 수 (활성/비활성/습득불가)	634	596 / 4 / 34
트리거 수 (활성/비활성 [장애/정상])	268	264 / 4 [0 / 264]
유저 수 (온라인)	2	1
1초간의 감시항목수(Zabbix서버의 요구 퍼포먼스)	5.92	

장애

시각	정보	호스트	장애 • 심각도	기간	인지	액션	태그
데이터를 찾을 수 없음.							

0 감시 중 0 미감시 5 알 수 없음 5 합계

0 심각한 장애 0 중증 장애 0 가벼운 장애 0 경고 0 정보 0 미분류

대시보드의 변경

즐겨찾기의 맵

맵이 추가되어 있지 않습니다.

AGENT

Control

Compute

Storage

MariaDB

Zabbix

1. Control 노드를 제외한 나머지 노드에 동일하게 적용
rpm -ivh http://repo.zabbix.com/zabbix/5.0/rhel/8/x86_64/zabbix-agent-5.0.1-1.el8.x86_64.rpm

2. Vi /etc/zabbix/zabbix_agentd.conf
119 Server=Zabbix Server IP(=Control IP)
144 StartAgents=0 (주석 제거 후 반드시 0으로 설정
0=Active, default=Passive)
160 ServerActive=Zabbix Server IP(=Control IP)
171 Hostname=hostname(작업 중인 해당 노드의 hostname
기입 ex. compute1)

3. systemctl enable zabbix-agent --now

HOST

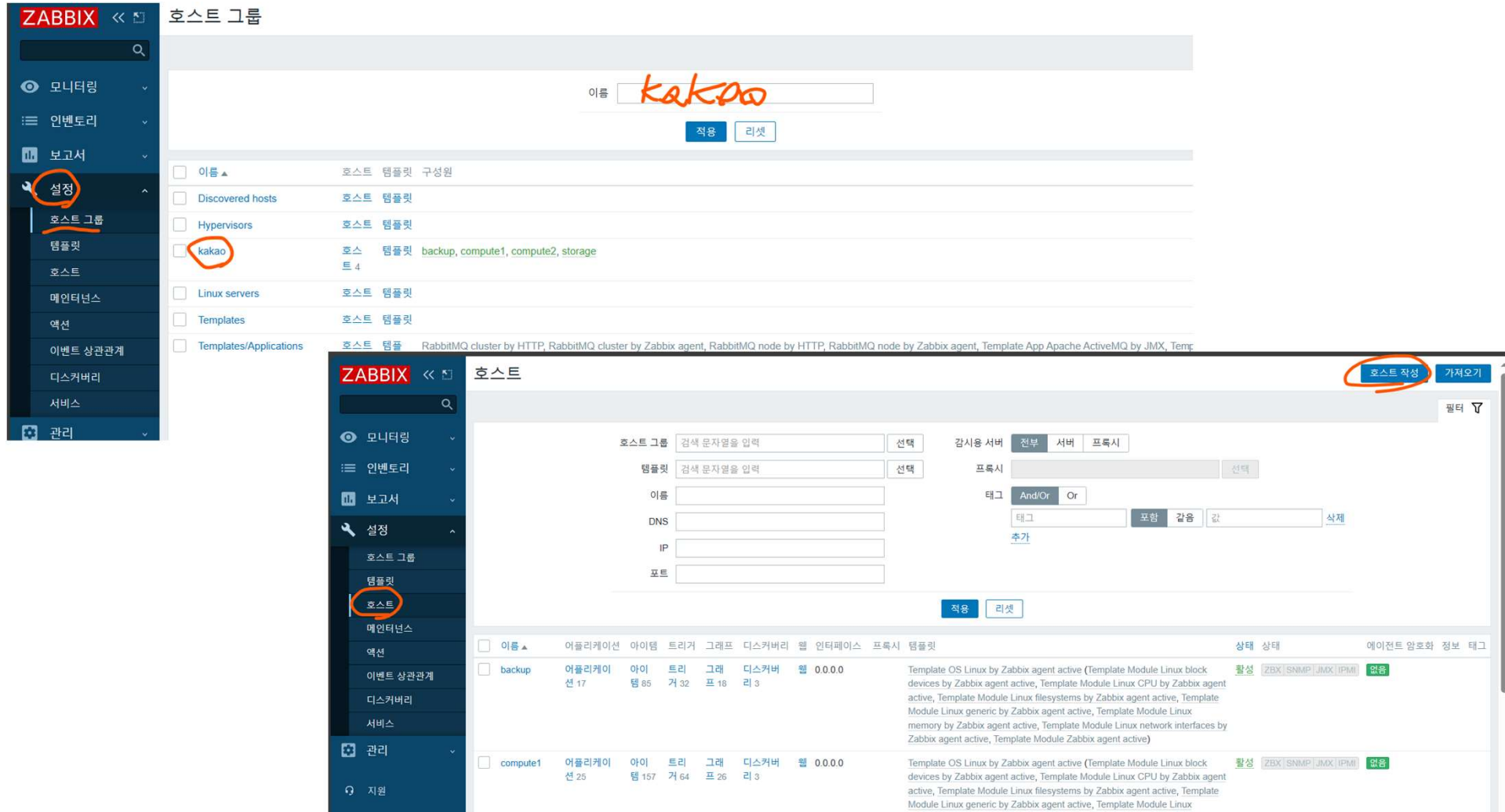
Control

Compute

Storage

MariaDB

Zabbix



The screenshot displays the Zabbix web interface for host management. The left sidebar contains navigation menus for Monitoring, Inventory, Reporting, Settings, Templates, and Maintenance. The 'Settings' menu is expanded, showing 'Host Groups' and 'Hosts'. The main content area is divided into two sections: 'Host Groups' and 'Hosts'.

Host Groups Section:

- Search bar: Name (kakao)
- Buttons: 적용 (Apply), 리셋 (Reset)
- Table of Host Groups:

이름	호스트	템플릿	구성원
Discovered hosts	호스트	템플릿	
Hypervisors	호스트	템플릿	
kakao	호스트	템플릿	backup, compute1, compute2, storage
Linux servers	호스트	템플릿	
Templates	호스트	템플릿	
Templates/Applications	호스트	템플	RabbitMQ cluster by HTTP, RabbitMQ cluster by Zabbix agent, RabbitMQ node by HTTP, RabbitMQ node by Zabbix agent, Template App Apache ActiveMQ by JMX, Tem

Hosts Section:

- Buttons: 호스트 작성 (Create Host), 가져오기 (Import)
- Form fields for host creation:

호스트 그룹	검색 문자열을 입력	선택	감시용 서버	전부	서버	프록시
템플릿	검색 문자열을 입력	선택	프록시			선택
이름			태그	And/Or	Or	
DNS			태그	포함	같음	없음
IP			추가			
포트						

Hosts Table:

이름	어플리케이션	아이템	트리거	그래프	디스커버리	웹	인터페이스	프록시	템플릿	상태	상태	에이전트 암호화	정보	태그
backup	어플리케이션	아이템	트리거	그래프	디스커버리	웹	0.0.0.0		Template OS Linux by Zabbix agent active (Template Module Linux block devices by Zabbix agent active, Template Module Linux CPU by Zabbix agent active, Template Module Linux filesystems by Zabbix agent active, Template Module Linux generic by Zabbix agent active, Template Module Linux memory by Zabbix agent active, Template Module Linux network interfaces by Zabbix agent active, Template Module Zabbix agent active)	활성				
compute1	어플리케이션	아이템	트리거	그래프	디스커버리	웹	0.0.0.0		Template OS Linux by Zabbix agent active (Template Module Linux block devices by Zabbix agent active, Template Module Linux CPU by Zabbix agent active, Template Module Linux filesystems by Zabbix agent active, Template Module Linux generic by Zabbix agent active, Template Module Linux	활성				

HOST

Control

Compute

Storage

MariaDB

Zabbix

호스트 탭들릿 IPMI 태그 매크로 인벤토리 암호화

* 호스트명 compute1

표시명 compute1

* 그룹 kakao ✕ 선택

검색 문자열을 입력

* Interfaces

종류	IP 주소	DNS 이름	접속 방법	포트	디플트
에이전트	0.0.0.0		IP DNS	0	삭제

추가

이름

프록시에 의한 감시 (프록시 없음) ▼

활성 ☒

추가 취소

호스트

호스트 탭들릿 IPMI 태그 매크로 인벤토리 암호화

템플릿과의 링크 이름 액션

Link new templates

Template OS Linux by Zabbix agent active ✕ 선택

검색 문자열을 입력

추가 취소

MONITOR

ZABBIX << >> 최근 데이터

모니터링 (red circle)

대쉬보드
장애
호스트
개요
최근 데이터
스크린
맵
디스커버리
서비스

인벤토리
보고서
설정
관리
지원

호스트 그룹: kakao (선택)
검색 문자열을 입력

호스트: compute1, compute2, backup, storage (선택)
검색 문자열을 입력

어플리케이션: (선택)

이름: (선택)

히스토리가 없는 아이템 보기: ☒
자세히 표시: ☐

적용 리셋

호스트	이름	최근 검사	최근 값	변화
backup	CPU (합계 17개)			
	Context switches per second	2024/01/08 13:49:37	276.7547	+4.5865 그래프
	CPU guest nice time	2024/01/08 13:49:37	0 %	그래프
	CPU guest time	2024/01/08 13:49:37	0 %	그래프
	CPU idle time	2024/01/08 13:49:37	99.1622 %	+0.03307 % 그래프
	CPU interrupt time	2024/01/08 13:49:37	0.2011 %	-0.01665 % 그래프
	CPU iowait time	2024/01/08 13:49:37	0.2178 %	-0.03339 % 그래프
	CPU nice time	2024/01/08 13:49:37	0 %	그래프
	CPU softirq time	2024/01/08 13:49:37	0.06702 %	+0.01678 % 그래프

1. 모니터링 – 최근 데이터에서 필터링으로 등록한 node들만 출력
2. 여러가지 물리자원 상태를 확인할 수 있다.

Weak Point

Control

```
#!/bin/bash

value=$(mysql kakaodb -u rep -ptest123 << EOF
select pub_ip from instance_info where name = 'centos1';
EOF
)
# echo "$value"
value=$(echo $value | tail -1 | gawk '{print $2}')
echo "$value"
```

8,0-1

Compute

Storage

```
192.168.1.99
Using username "root".
root@192.168.1.99's password:
Last login: Mon Jan  8 16:12:12 2024 from 192
[root@control ~]# ./test.sh
NULL
```

MariaDB

Zabbix

```
pub_ip=$(echo $(hostname -I) | gawk '{print $1}')
pri_ip=$(echo $(hostname -I) | gawk '{print $2}')

echo "pub_ip=$pub_ip"
echo "pri_ip=$pri_ip"

mysql -h 192.168.1.99 kakaodb -u rep -ptest123 << EOF
insert into instance_info(pub_ip, pri_ip) values('$pub_ip', '$pri_ip');
EOF
```

Kakao cloud

감사합니다.