

—
설치 및 환경설정
—

TAS-TAC 구축 및 가상환경(VMware) 구성방법

TMAXTibero

Copyright © 2025 TmaxTibero. All Rights Reserved

Copyright Notice

Copyright © 2025 TIBERO Co., Ltd. All Rights Reserved.

대한민국 경기도 성남시 분당구 정자일로 45 티맥스소프트타워

Website

www.tmaxtibero.com

Restricted Rights Legend

All TIBERO Software (Tibero®) and documents are protected by copyright laws and international convention. TIBERO software and documents are made available under the terms of the TIBERO License Agreement and may only be used or copied in accordance with the terms of this agreement. No part of this document may be transmitted, copied, deployed, or reproduced in any form or by any means, electronic, mechanical, or optical, without the prior written consent of TIBERO Co., Ltd.

이 소프트웨어(Tibero®) 사용설명서의 내용과 프로그램은 저작권법과 국제 조약에 의해서 보호받고 있습니다. 사용설명서의 내용과 여기에 설명된 프로그램은 TIBERO Co., Ltd.와의 사용권 계약 하에서만 사용이 가능하며, 사용권 계약을 준수하는 경우에만 사용 또는 복제할 수 있습니다. 이 사용설명서의 전부 또는 일부분을 TIBERO의 사전 서면 동의 없이 전자, 기계, 녹음 등의 수단을 사용하여 전송, 복제, 배포, 2차적 저작물 작성 등의 행위를 하여서는 안 됩니다.

Trademarks

Tibero® is a registered trademark of TIBERO Co., Ltd. Other products, titles or services may be registered trademarks of their respective companies.

Tibero®는 TIBERO Co., Ltd.의 등록 상표입니다. 기타 모든 제품들과 회사 이름은 각각 해당 소유주의 상표로서 참조용으로만 사용됩니다.

안내서 정보

안내서 제목: TAS-TAC 구축 및 가상환경(VMware) 구성방법

발행일: 2025-12-23

소프트웨어 버전: Tibero 7.2.4

안내서 버전: 1.0.0

제, 개정 이력

안내서 버전	개정일자	개정 사유 및 내용	비고
1.0.0	2025.12.23	최초 제정	

목차

0. TAS-TAC 환경 설정	4
(1) 버전	4
(2) 네트워크(IP/PORT) 및 SID 구성.....	4
(3) 스토리지 구성	5
1. 가상환경(VMware) 서버 및 공유디스크 구성	6
(1) 서버 생성 완료.....	6
(2) 1번 서버 - 공유 디스크 생성	6
(3) 2번 서버 - 공유 디스크 생성	8
(4) 공유 디스크 설정 변경.....	9
(5) 서버 기동 후, 디스크 확인.....	10
2. 설치 사전 작업	12
(1) 커널 파라미터 설정.....	12
(2) 공유 디스크 udev 구성.....	12
(3) 계정 생성 및 profile.....	14
(4) 바이너리 압축해제 및 라이선스 적용	15
(5) 파라미터 파일 생성 및 설정 작업	16
3. TAS-TAC 설치	19
(1) 1번 노드 - TAS Diskspace 생성.....	19
(2) 1번 노드 - CM 설치 및 Diskspace, thread 추가.....	19
(3) 1번 노드 - DB 생성	21
(4) 1번 노드 - 시스템 오브젝트 생성.....	23
(5) 2번 노드 - CM 설치.....	23
(6) 2번 노드 - TAS 인스턴스 설치.....	24
(7) 2번 노드 - DB 인스턴스 설치.....	25
(8) 설치 및 동기화 확인	25

0. TAS-TAC 환경 설정

본 문서에는 Tibero 설치를 위한 OS 설치, 패키지 설치 및 커널 파라미터 설정 등에 대해서는 기술하고 있지 않습니다. 해당 작업들은 모두 사전에 진행하였다는 가정 하에 작성되었습니다.

(1) 버전

- OS: Rocky Linux release 9.6 (Blue Onyx) / 5.14.0-570.17.1.el9_6.x86_64
- Tibero: Tibero 7 (DB 7.2.4) Build 303837

(2) 네트워크(IP/PORT) 및 SID 구성

- Server

항목	Node 1	Node 2
Hostname	server1	server2
OS User	tibero	tibero
Public IP	192.168.92.128 (ens160)	192.168.92.129 (ens160)

- CM

항목	Node 1	Node 2
SID	cm1	cm2
CM Inter-Connect Port	11019	11019
CM_UI_PORT	11039	11039

- TAS

항목	Node 1	Node 2
SID	tas1	tas2
LISTENER_PORT	9629	9629
LOCAL_CLUSTER_ADDR	192.168.182.128 (ens192)	192.168.182.129 (ens192)
LOCAL_CLUSTER_PORT	11049	11049
CM_PORT	11039	11039

- Tibero

항목	Node 1	Node 2
SID	tibero1	tibero2
DB_NAME	tiberodb	
LISTENER_PORT	8629	8629
LOCAL_CLUSTER_ADDR	192.168.182.128 (ens192)	192.168.182.129 (ens192)
LOCAL_CLUSTER_PORT	11029	11029
CM_PORT	11039	11039
AS_PORT	9629	9629

(3) 스토리지 구성

리소스 설정은 실습 환경에 맞게 조절 가능합니다.

항목	Node 1	Node 2	비고
로컬 디스크	Disk 1 (20G)	Disk 1 (20G)	각 노드의 엔진/로그 영역
공유 디스크 1		Disk 2 (10G)	SYSTEM 디스크스페이스
공유 디스크 2		Disk 3 (10G)	ARCH 디스크스페이스
공유 디스크 3		Disk 4 (10G)	DATA 디스크스페이스
공유 디스크 4		Disk 5 (10G)	DATA 디스크스페이스

1. 가상환경(VMware) 서버 및 공유디스크 구성

(1) 서버 생성 완료

- 메모리: 8G
- 프로세서: 1CPU * 2Core * 1Thread
- 네트워크 어댑터: 1) Public IP (NAT) 2) Inter-Connect IP (Host-Only)
- 로컬디스크: 20G

```
[root@server1 ~]# free -h
total        used        free        shared    buff/cache   available
Mem:          7.7Gi       2.7Gi       4.5Gi       1.9Gi       2.7Gi       5.0Gi
Swap:         2.0Gi          0B           2.0Gi
[root@server1 ~]# lscpu | grep -E 'Socket(s)\|Core(s)\|Thread(s)\|
Thread(s) per core: 1
Core(s) per socket: 2
Socket(s): 1
[root@server1 ~]# ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.92.128 netmask 255.255.255.0 broadcast 192.168.92.255
    inet6 fe80::20c:29ff:fe7e:862a prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:7e:86:2a txqueuelen 1000 (Ethernet)
    RX packets 34068 bytes 2473676 (2.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 58890 bytes 13530194 (12.9 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens192: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.182.128 netmask 255.255.255.0 broadcast 192.168.182.255
    inet6 fe80::20c:29ff:fe7e:8634 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:7e:86:34 txqueuelen 1000 (Ethernet)
    RX packets 405399 bytes 60178490 (57.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 339550 bytes 136109628 (129.8 MiB)
    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 107959 bytes 20006916 (19.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 107959 bytes 20006916 (19.0 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@server1 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
srd0 11:0 1 1024M 0 rom
nvme0n1 259:0 0 20G 0 disk
├─nvme0n1p1 259:1 0 1G 0 part /boot
├─nvme0n1p2 259:2 0 19G 0 part
├─rl-root 253:0 0 17G 0 lvm /
└─rl-swap 253:1 0 2G 0 lvm [SWAP]
```

```
[root@server2 ~]# free -h
total        used        free        shared    buff/cache   available
Mem:          7.7Gi       2.7Gi       4.8Gi       1.9Gi       2.4Gi       5.1Gi
Swap:         2.0Gi          0B           2.0Gi
[root@server2 ~]# lscpu | grep -E 'Socket(s)\|Core(s)\|Thread(s)\|
Thread(s) per core: 1
Core(s) per socket: 2
Socket(s): 1
[root@server2 ~]# ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.92.129 netmask 255.255.255.0 broadcast 192.168.92.255
    inet6 fe80::20c:29ff:fe6d:e7a1 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:6d:e7:a1 txqueuelen 1000 (Ethernet)
    RX packets 6052 bytes 596532 (582.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5895 bytes 592453 (578.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens192: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.182.129 netmask 255.255.255.0 broadcast 192.168.182.255
    inet6 fe80::20c:29ff:fe6d:e7ab prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:6d:e7:ab txqueuelen 1000 (Ethernet)
    RX packets 340038 bytes 137530459 (131.1 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 406186 bytes 58696448 (55.9 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

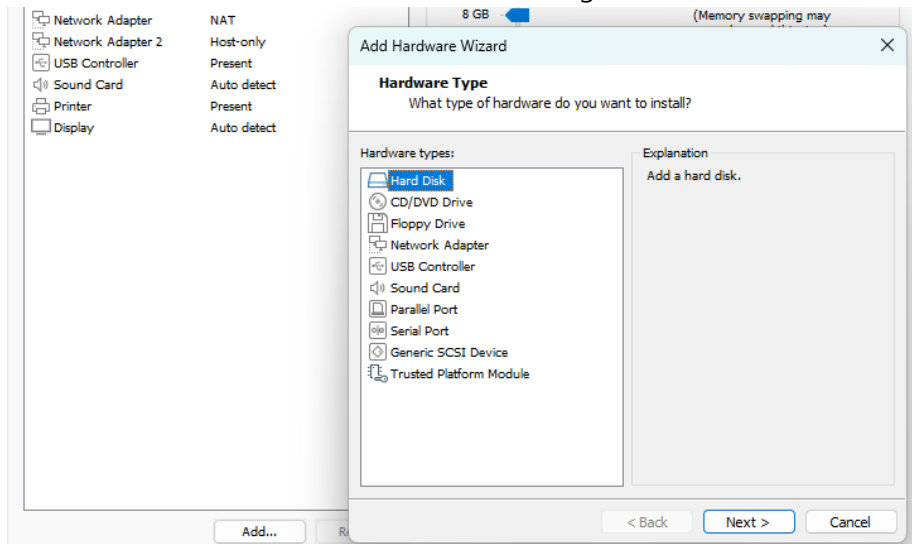
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 922 bytes 205376 (200.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 922 bytes 205376 (200.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@server2 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
srd0 11:0 1 1024M 0 rom
nvme0n1 259:0 0 20G 0 disk
├─nvme0n1p1 259:1 0 1G 0 part /boot
├─nvme0n1p2 259:2 0 19G 0 part
├─rl-root 253:0 0 17G 0 lvm /
└─rl-swap 253:1 0 2G 0 lvm [SWAP]
```

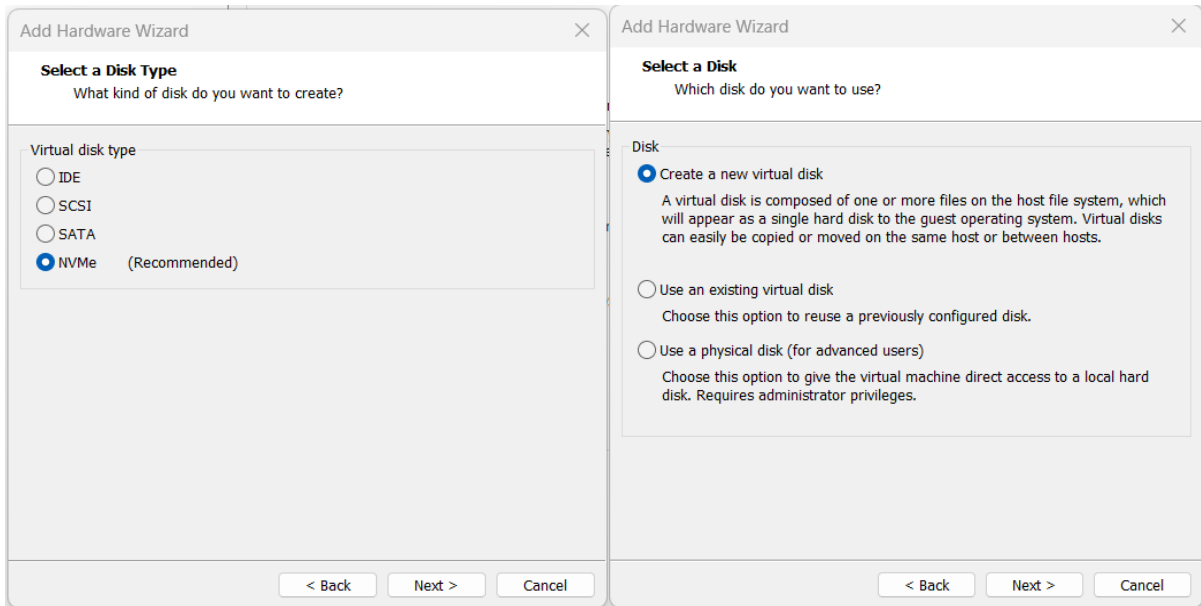
1번, 2번 서버를 위와 같이 생성하였다는 가정 하에 이후 단계를 진행하면 됩니다.
리소스 설정은 실습 환경에 맞게 조절 가능합니다.

(2) 1번 서버 - 공유 디스크 생성

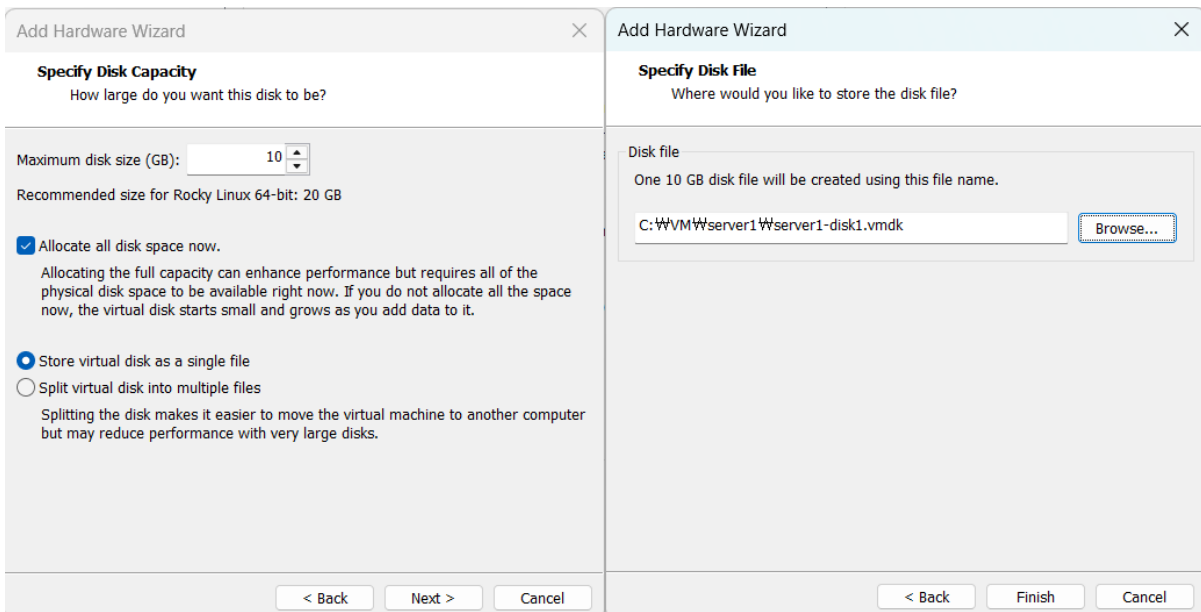
- 1번 서버에 하드 디스크를 추가합니다. Setting -> Add -> Hard Disk 선택



- 다음 선택창에서 Disk Type -> Disk -> Disk 용량 -> Disk File 명을 지정하여 생성합니다.



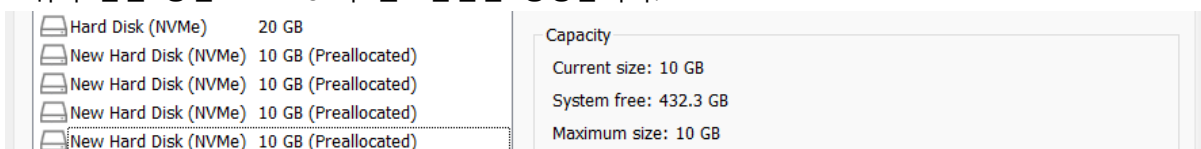
* 1번에서는 **Create a new virtual disk**. 2번에서는 **Use an existing virtual disk**를 선택합니다.



* 공유 공간을 할당하기 위해서는 **Allocate all disk space now** 체크박스를 선택하여 실제 사용할 디스크 크기만큼 미리 최대치로 가상 디스크 이미지를 생성하고, **Store virtual disk as a single file** 옵션으로 여러 개 파일이 아닌 단일 파일로 디스크 이미지를 생성합니다.

(사진과 같이 10G 할당 시, 10G 크기의 server1-disk1 파일 생성)

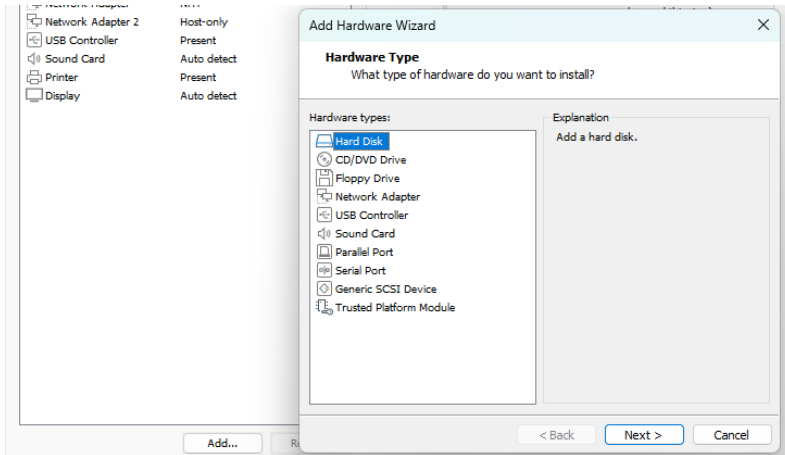
- 위와 같은 방법으로 Disk가 필요한만큼 생성합니다.



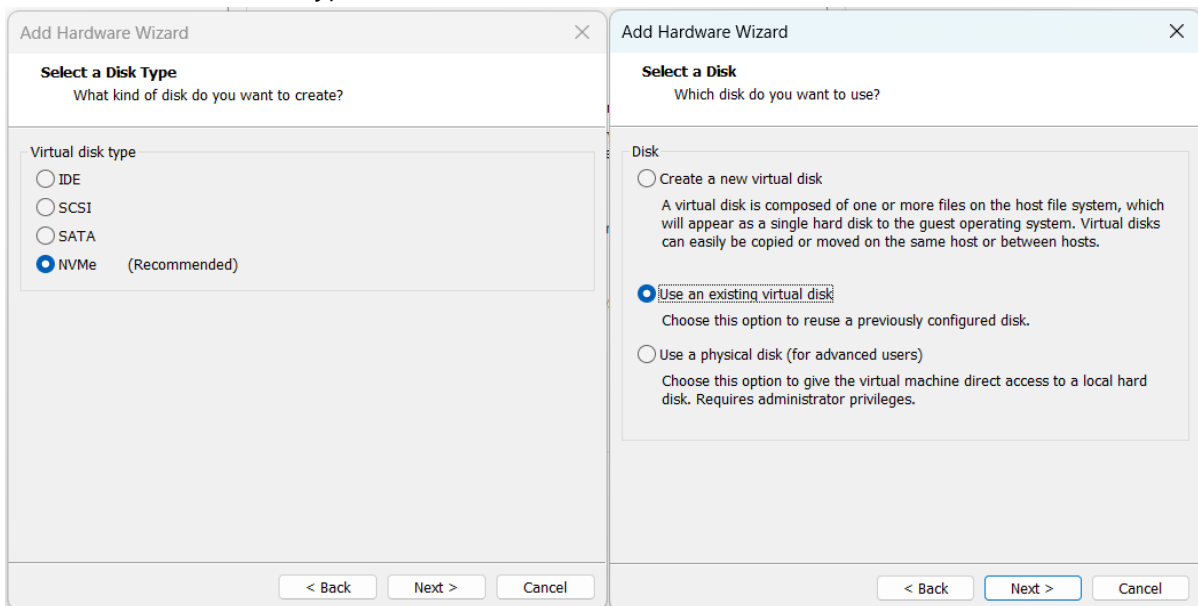
* 1번 서버에 디스크를 추가한 화면

(3) 2번 서버 - 공유 디스크 생성

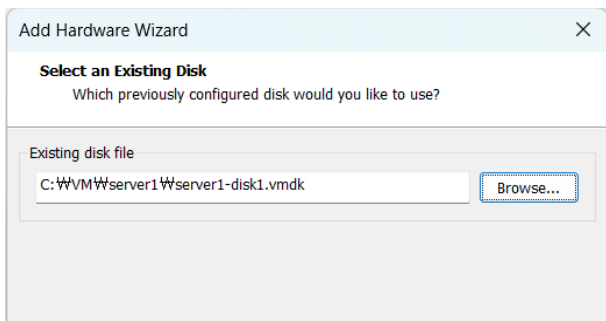
- 2번 서버에 하드 디스크를 추가합니다. Setting -> Add -> Hard Disk 선택



- 다음 선택창에서 Disk Type -> Disk -> Disk 용량 -> Disk File 명을 지정하여 생성합니다.

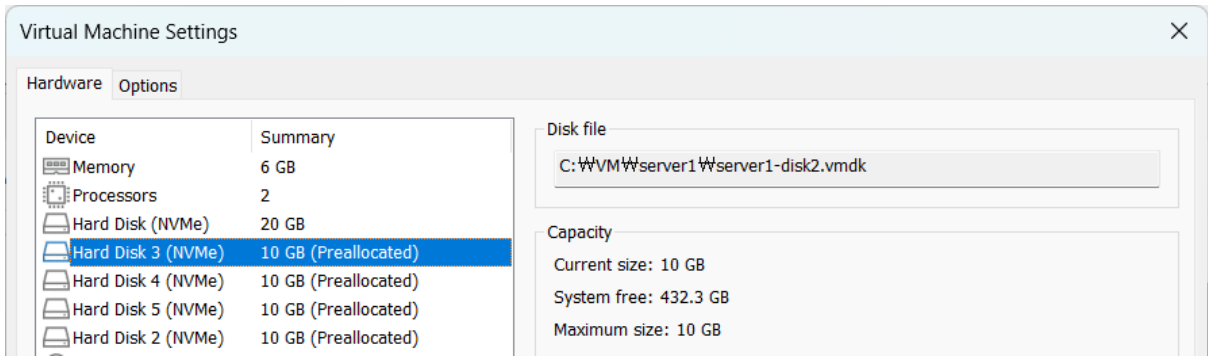


* 1번에서는 **Create a new virtual disk**. 2번에서는 **Use an existing virtual disk**를 선택합니다.



* 1번에서 만들어준 Disk file을 선택해서 만들어줍니다.

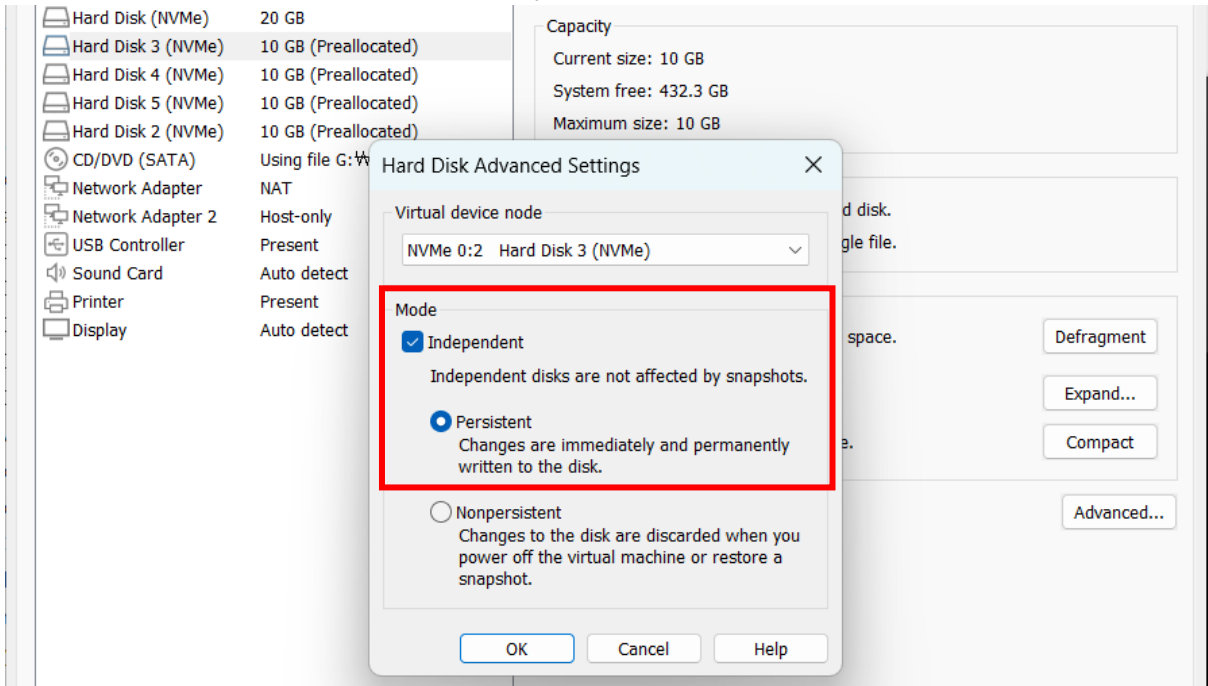
- 위와 같은 방법으로 Disk 수만큼 생성합니다.



* 2번 서버에 디스크를 추가한 화면

(4) 공유 디스크 설정 변경

- 양 서버에서 모든 디스크의 설정을 아래와 같이 변경해줍니다. Setting -> Hard Disk 선택 -> Advanced 선택 -> 아래와 같이 변경 (Independent, Persistent 체크)



- 사진과 같이 양 쪽 서버에서 Persistent 디스크 확인

Device	Summary	Device	Summary
Memory	8 GB	Memory	8 GB
Processors	2	Processors	2
Hard Disk (NVMe)	20 GB	Hard Disk (NVMe)	20 GB
Hard Disk 2 (NVMe)	10 GB (Persistent)	Hard Disk 3 (NVMe)	10 GB (Persistent)
Hard Disk 3 (NVMe)	10 GB (Persistent)	Hard Disk 4 (NVMe)	10 GB (Persistent)
Hard Disk 4 (NVMe)	10 GB (Persistent)	Hard Disk 5 (NVMe)	10 GB (Persistent)
Hard Disk 5 (NVMe)	10 GB (Persistent)	Hard Disk 2 (NVMe)	10 GB (Persistent)
CD/DVD (SATA)	Using file G:\#WVM\Rocky-9.6-x...	CD/DVD (SATA)	Using file G:\#WVM\Rocky-9.6-x...
Network Adapter	NAT	Network Adapter	NAT
Network Adapter 2	Host-only	Network Adapter 2	Host-only
USB Controller	Present	USB Controller	Present
Sound Card	Auto detect	Sound Card	Auto detect
Printer	Present	Printer	Present
Display	Auto detect	Display	Auto detect

- 양 서버의 vmx 파일을 텍스트 에디터 프로그램으로 열어 편집합니다.

▼ Virtual Machine Details

State: Powered off
Configuration file: C:\VM\server1\#tas1.vmx
 Hardware compatibility: Workstation 17.5 or later virtual machine
 Primary IP address: Network information is not available

▼ Virtual Machine Details

State: Powered off
Configuration file: C:\VM\server2\#tas2.vmx
 Hardware compatibility: Workstation 17.5 or later virtual machine
 Primary IP address: Network information is not available

* 왼쪽: 1번 서버 / 오른쪽: 2번 서버

- 아래와 같이 파일을 수정합니다. nvme0:1~4 와 같은 디바이스 노드 이름은 실습 환경에 맞게 변경하여 사용합니다.

```
disk.locking="FALSE"
diskLib.dataCacheMaxSize="0"
nvme0:1.fileName = "server1-disk1.vmdk"
nvme0:2.fileName = "server1-disk2.vmdk"
nvme0:3.fileName = "server1-disk3.vmdk"
nvme0:4.fileName = "server1-disk4.vmdk"
nvme0:1.sharedbus="virtual"
nvme0:2.sharedbus="virtual"
nvme0:3.sharedbus="virtual"
nvme0:4.sharedbus="virtual"
nvme0:1.deviceType="disk"
nvme0:2.deviceType="disk"
nvme0:3.deviceType="disk"
nvme0:4.deviceType="disk"
```

```
115 disk.locking="FALSE"
116 diskLib.dataCacheMaxSize="0"
117 nvme0:1.fileName = "C:\VM\server1\server1-disk1.vmdk"
118 nvme0:2.fileName = "C:\VM\server1\server1-disk2.vmdk"
119 nvme0:3.fileName = "C:\VM\server1\server1-disk3.vmdk"
120 nvme0:4.fileName = "C:\VM\server1\server1-disk4.vmdk"
121 nvme0:1.sharedbus="virtual"
122 nvme0:2.sharedbus="virtual"
123 nvme0:3.sharedbus="virtual"
124 nvme0:4.sharedbus="virtual"
125 nvme0:1.deviceType="disk"
126 nvme0:2.deviceType="disk"
127 nvme0:3.deviceType="disk"
128 nvme0:4.deviceType="disk"
```

* 왼쪽: 1번 서버 / 오른쪽: 2번 서버

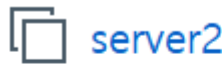
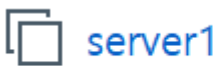
* disk.locking="FALSE" / diskLib.dataCacheMaxSize="0" 를 추가해줍니다.

* nvme0:1~4.fileName 은 양 서버에서 같은 디스크 파일을 바라보도록 설정합니다.

* nvme0:1~4.sharedbus="virtual" / nvme0:1~4.deviceType="disk" 를 추가해줍니다.

(5) 서버 기동 후, 디스크 확인

- 설정 완료 후, 양 쪽 서버 기동



▶ Power on this virtual machine
 Edit virtual machine settings

▶ Power on this virtual machine
 Edit virtual machine settings

* 왼쪽: 1번 서버 / 오른쪽: 2번 서버

- 정상 기동 후, 양 서버에서 df 와 lsblk 로 파일시스템과 디스크 구조 확인

```
[root@server1 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M  0  4.0M  0% /dev
tmpfs           2.9G  0  2.9G  0% /dev/shm
tmpfs           1.2G  9.1M  1.2G  1% /run
/dev/mapper/rl-root 17G  8.8G  8.2G  52% /
/dev/nvme0n1p1  960M 290M  671M  31% /boot
tmpfs           589M  4.0K  589M  1% /run/user/0

[root@server1 ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sr0         11:0    1 1024M  0 rom
nvme0n1     259:0    0   20G  0 disk
├─nvme0n1p1 259:1    0    1G  0 part /boot
├─nvme0n1p2 259:2    0   19G  0 part
├─rl-root   253:0    0   17G  0 lvm /
├─rl-swap  253:1    0    2G  0 lvm [SWAP]
nvme0n2     259:3    0   10G  0 disk
nvme0n3     259:4    0   10G  0 disk
nvme0n4     259:5    0   10G  0 disk
nvme0n5     259:6    0   10G  0 disk

[root@server1 ~]#

[root@server2 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M  0  4.0M  0% /dev
tmpfs           2.9G  0  2.9G  0% /dev/shm
tmpfs           1.2G  9.1M  1.2G  1% /run
/dev/mapper/rl-root 17G  7.8G  9.3G  46% /
/dev/nvme0n1p1  960M 290M  671M  31% /boot
tmpfs           589M  4.0K  589M  1% /run/user/0

[root@server2 ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sr0         11:0    1 1024M  0 rom
nvme0n1     259:0    0   20G  0 disk
├─nvme0n1p1 259:1    0    1G  0 part /boot
├─nvme0n1p2 259:2    0   19G  0 part
├─rl-root   253:0    0   17G  0 lvm /
├─rl-swap  253:1    0    2G  0 lvm [SWAP]
nvme0n2     259:3    0   10G  0 disk
nvme0n3     259:4    0   10G  0 disk
nvme0n4     259:5    0   10G  0 disk
nvme0n5     259:6    0   10G  0 disk

[root@server2 ~]#
```

- fdisk 로 공유디스크 양 서버 인식여부 확인

```
[root@server1 ~]# fdisk -l
Disk /dev/nvme0n1: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x0822ae9c

Device      Boot  Start      End  Sectors  Size Id Type
/dev/nvme0n1p1 *    2048    2099199    2097152    1G 83 Linux
/dev/nvme0n1p2      2099200 41943039 39843840    19G 8e Linux LVM

Disk /dev/nvme0n2: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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/nvme0n4: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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/nvme0n5: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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/rl-root: 17 GiB, 18249416704 bytes, 35643392 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/rl-swap: 2 GiB, 2147483648 bytes, 4194304 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@server1 ~]#
```

```
[root@server2 ~]# fdisk -l
Disk /dev/nvme0n1: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x3c563bfd

Device      Boot  Start      End  Sectors  Size Id Type
/dev/nvme0n1p1 *    2048    2099199    2097152    1G 83 Linux
/dev/nvme0n1p2      2099200 41943039 39843840    19G 8e Linux LVM

Disk /dev/nvme0n2: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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/nvme0n4: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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/nvme0n5: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VMware Virtual NVMe Disk
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/rl-root: 17 GiB, 18249416704 bytes, 35643392 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/rl-swap: 2 GiB, 2147483648 bytes, 4194304 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@server2 ~]#
```

* 왼쪽: 1번 서버 / 오른쪽: 2번 서버

2. 설치 사전 작업

(1) 커널 파라미터 설정

- 커널 파라미터를 아래와 같이 수정

=> `vi /etc/sysctl.conf` 진입하여 `fs.aio-max-nr=4194304` 적용

=> `sysctl -p` 명령어로 변경사항 적용

```
# sysctl settings are defined through files in
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.
#
# Vendors settings live in /usr/lib/sysctl.d/.
# To override a whole file, create a new file with the same in
# /etc/sysctl.d/ and put new settings there. To override
# only specific settings, add a file with a lexically later
# name in /etc/sysctl.d/ and put new settings there.
#
# For more information, see sysctl.conf(5) and sysctl.d(5).

kernel.sem=10000 32000 10000 10000
kernel.shmmax=4144564224
kernel.shmall=1011857
kernel.shmmni=4096

fs.file-max=67108864
#fs.aio-max-nr=1048576
fs.aio-max-nr=4194304
fs.nr_open=3000000

net.ipv4.ip_local_port_range=1024 65500
net.core.rmem_default = 262144
net.core.wmem_default = 262144
net.core.rmem_max = 67108864
net.core.wmem_max = 67108864
```

* TAS를 이용해 Tiberio를 구성하는 경우, 기존 커널 파라미터 설정에서 `fs.aio-max-nr` 값 증설이 필요합니다.
(1048576 -> 4194304)

(2) 공유 디스크 udev 구성

- 공유 디스크 ID 확인 => `ls -al /dev/disk/by-id | grep nvme`

* 디스크 ID로 디스크 링크를 해주어야, OS 재기동 시 디스크 순서가 바뀌어도 문제가 생기지 않습니다.

* 본 문서에서는 nvme로 디스크를 생성하여 nvme로 조회하였으나, 실습 환경에 맞게 해당 경로에서 디스크 ID를 검색합니다.

* 본 문서에서는 `ENV{ID_WWN}` 값으로 매핑하였지만, 실습 환경에 따라 `ID_SERIAL` 값이 고유한 값인 경우 `ENV{ID_SERIAL}`으로 디스크 매핑을 해주어도 됩니다.

```
[root@server1 by-id]# pwd
/dev/disk/by-id
[root@server1 by-id]# ls -al | grep nvme
lrwxrwxrwx. 1 root root 15 Nov 9 22:09 lvm-pv-uuid-iPt3d8-Cp7S-R640-0t4C-rGr7-1dcC-b1760M -> /nvme0n1p2
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-eui.01d9b2000267a30d000c296a15600465 -> ../../nvme0n4
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-eui.297f434967caf19d000c296b4073616f -> ../../nvme0n5
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-eui.3549da3825a3ae5d000c2961e15ad0d0 -> ../../nvme0n3
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-eui.7421b5814d9b3a14000c296eae8b24a2 -> ../../nvme0n2
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-eui.8c91de1985f9cb8e000c296f23371ab9 -> ../../nvme0n1
lrwxrwxrwx. 1 root root 15 Nov 9 22:09 nvme-eui.8c91de1985f9cb8e000c296f23371ab9-part1 -> ../../nvme0n1p1
lrwxrwxrwx. 1 root root 15 Nov 9 22:09 nvme-eui.8c91de1985f9cb8e000c296f23371ab9-part2 -> ../../nvme0n1p2
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000 -> ../../nvme0n2
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_1 -> ../../nvme0n1
lrwxrwxrwx. 1 root root 15 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_1-part1 -> ../../nvme0n1p1
lrwxrwxrwx. 1 root root 15 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_1-part2 -> ../../nvme0n1p2
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_2 -> ../../nvme0n2
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_3 -> ../../nvme0n3
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_4 -> ../../nvme0n4
lrwxrwxrwx. 1 root root 13 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_5 -> ../../nvme0n5
lrwxrwxrwx. 1 root root 15 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000-part1 -> ../../nvme0n1p1
lrwxrwxrwx. 1 root root 15 Nov 9 22:09 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000-part2 -> ../../nvme0n1p2
[root@server1 by-id]#
```

* 1번 서버에서 확인한 결과

```
[root@server2 by-id]# pwd
/dev/disk/by-id
[root@server2 by-id]# ls -al | grep nvme
lrwxrwxrwx. 1 root root 15 Nov 9 22:10 lvm-pv-uuid-G2Ulr02-yv72-Pfap-mVbE-ey0u-4p7d-xqTczi -> /nvme0n1p2
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-eui.01d9b2000267a30d000c296a15600465 -> ../../nvme0n4
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-eui.297f434967caf19d000c296b4073616f -> ../../nvme0n5
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-eui.3549da3825a3ae5d000c2961e15ad0d0 -> ../../nvme0n3
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-eui.7421b5814d9b3a14000c296eae8b24a2 -> ../../nvme0n2
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-eui.cbdb43e45004745a000c296409c0f7ce -> ../../nvme0n1
lrwxrwxrwx. 1 root root 15 Nov 9 22:10 nvme-eui.cbdb43e45004745a000c296409c0f7ce-part1 -> ../../nvme0n1p1
lrwxrwxrwx. 1 root root 15 Nov 9 22:10 nvme-eui.cbdb43e45004745a000c296409c0f7ce-part2 -> ../../nvme0n1p2
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000 -> ../../nvme0n3
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_1 -> ../../nvme0n1
lrwxrwxrwx. 1 root root 15 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_1-part1 -> ../../nvme0n1p1
lrwxrwxrwx. 1 root root 15 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_1-part2 -> ../../nvme0n1p2
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_2 -> ../../nvme0n2
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_3 -> ../../nvme0n3
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_4 -> ../../nvme0n4
lrwxrwxrwx. 1 root root 13 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000_5 -> ../../nvme0n5
lrwxrwxrwx. 1 root root 15 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000-part1 -> ../../nvme0n1p1
lrwxrwxrwx. 1 root root 15 Nov 9 22:10 nvme-VMware_Virtual_NVMe_Disk_VMware_NVME_0000-part2 -> ../../nvme0n1p2
[root@server2 by-id]#
```

* 2번 서버에서 확인한 결과

- udev rules 파일로 공유 디스크에 링크 생성

=> vi /etc/udev/rules.d/tas.rules (rule 파일명은 임의 지정 가능) 진입하여 디스크 ID와 매핑 된 tiberio 유저 소유의 링크 지정

* ENV{ID_WWN}== : Disk ID 입력 / SYMLINK+= : 디스크 링크명 입력

=> udevadm control --reload-rules => udevadm trigger 명령어로 적용

- 생성된 링크를 확인 => ls -alrt /dev/tas(링크 경로)

```
[root@server1 rules.d]# cat tas.rules
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.7421b5814d9b3a14000c296eae8b24a2", SYMLINK+="tas/disk01", OWNER="tiberio", GROUP="dba", MODE="0660"
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.3549da3825a3ae5d000c2961e15ad0d0", SYMLINK+="tas/disk02", OWNER="tiberio", GROUP="dba", MODE="0660"
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.01d9b2000267a30d000c296a15600465", SYMLINK+="tas/disk03", OWNER="tiberio", GROUP="dba", MODE="0660"
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.297f434967caf19d000c296b4073616f", SYMLINK+="tas/disk04", OWNER="tiberio", GROUP="dba", MODE="0660"
[root@server1 rules.d]# udevadm control --reload-rules
[root@server1 rules.d]# udevadm trigger
[root@server1 rules.d]# cd /dev/tas
[root@server1 tas]# ls -alrt
total 0
lrwxrwxrwx. 1 root root 10 Nov 9 23:30 disk04 -> ../../nvme0n5
lrwxrwxrwx. 1 root root 10 Nov 9 23:30 disk03 -> ../../nvme0n4
lrwxrwxrwx. 1 root root 10 Nov 9 23:30 disk02 -> ../../nvme0n3
lrwxrwxrwx. 1 root root 10 Nov 9 23:30 disk01 -> ../../nvme0n2
drwxr-xr-x. 2 root root 120 Nov 9 23:30 .
drwxr-xr-x. 21 root root 3520 Nov 9 23:30 ..
[root@server1 tas]#
```

* 1번 서버에서 확인한 결과. 링크의 소유권은 root지만, 디스크 소유권은 tiberio임.

```
[root@server2 rules.d]# cat tas.rules
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.7421b5814d9b3a14000c296ae8b24a2", SYMLINK+="tas/disk01", OWNER="tibero", GROUP="dba", MODE="0660"
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.13549da3825a3ae5d000c2961e15ad0d0", SYMLINK+="tas/disk02", OWNER="tibero", GROUP="dba", MODE="0660"
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.01d9b2000267a30d000c296a15600465", SYMLINK+="tas/disk03", OWNER="tibero", GROUP="dba", MODE="0660"
KERNEL=="nvme0n?", ENV{ID_WWN}=="eui.297f434967caf19d000c296b4073616f", SYMLINK+="tas/disk04", OWNER="tibero", GROUP="dba", MODE="0660"
[root@server2 rules.d]# udevadm control --reload-rules
[root@server2 rules.d]# udevadm trigger
[root@server2 rules.d]# cd /dev/tas
[root@server2 tas]# ls -arlt
total 0
lrwxrwxrwx. 1 root root 10 Nov 9 23:31 disk02 -> ../nvme0n3
lrwxrwxrwx. 1 root root 10 Nov 9 23:31 disk04 -> ../nvme0n5
lrwxrwxrwx. 1 root root 10 Nov 9 23:31 disk01 -> ../nvme0n2
lrwxrwxrwx. 1 root root 10 Nov 9 23:31 disk03 -> ../nvme0n4
drwxr-xr-x. 2 root root 120 Nov 9 23:31 .
drwxr-xr-x. 21 root root 3520 Nov 9 23:31 ..
[root@server2 tas]#
```

* 2번 서버에서 확인한 결과. 링크의 소유권은 root지만, 디스크 소유권은 tibero임.

(3) 계정 생성 및 profile

- dba 그룹 및 tibero 계정 생성

=> **groupadd dba**

=> **useradd -g dba tibero**

- 기본 profile인 .bash_profile, CM에서 TAS 인스턴스 등록 및 실행시에 필요한 profile(이름 임의 지정 가능), CM에서 TAC(Tibero) 인스턴스 등록 및 실행시에 필요한 profile(이름 임의 지정 가능) 총 3개의 profile을 구성합니다.

① .bash_profile(1번 노드) / .bash_profile(2번 노드)

<pre>PATH=\$PATH:\$HOME/.local/bin:\$HOME/bin export PATH #----- # profile ENV #----- umask 027 set -o vi stty erase ^H export EDITOR=vi export TERM=vt100 export PS1="[LOGNAME@`hostname`:\\$TB_SID:\\$PWD]" #----- # TIBERO ENV #----- echo "Select TB_SID:" select sid in tas1 tibero1; do case \$sid in tas1 tibero1) export TB_SID=\$sid echo "TB_SID is set to \$TB_SID" break ;; *) echo "Invalid selection. Try again." ;; esac done tas() { export TB_SID=tas1 } tibero() { export TB_SID=tibero1 } export TB_HOME=/home/tibero/tibero7 export TB_PROF_DIR=\$TB_HOME/bin/prof export PATH=.:\$TB_HOME/bin:\$TB_HOME/client/bin:\$PATH export LD_LIBRARY_PATH=\$TB_HOME/lib:\$TB_HOME/client/lib:\$LD_LI BRARY_PATH</pre>	<pre>PATH=\$PATH:\$HOME/.local/bin:\$HOME/bin export PATH #----- # profile ENV #----- umask 027 set -o vi stty erase ^H export EDITOR=vi export TERM=vt100 export PS1="[LOGNAME@`hostname`:\\$TB_SID:\\$PWD]" #----- # TIBERO ENV #----- echo "Select TB_SID:" select sid in tas2 tibero2; do case \$sid in tas2 tibero2) export TB_SID=\$sid echo "TB_SID is set to \$TB_SID" break ;; *) echo "Invalid selection. Try again." ;; esac done tas() { export TB_SID=tas2 } tibero() { export TB_SID=tibero2 } export TB_HOME=/home/tibero/tibero7 export TB_PROF_DIR=\$TB_HOME/bin/prof export PATH=.:\$TB_HOME/bin:\$TB_HOME/client/bin:\$PATH export LD_LIBRARY_PATH=\$TB_HOME/lib:\$TB_HOME/client/lib:\$LD_LI BRARY_PATH</pre>
--	--

<pre>#----- # CM ENV #----- export CM_HOME=\$TB_HOME export CM_SID=cm1</pre>	<pre>#----- # CM ENV #----- export CM_HOME=\$TB_HOME export CM_SID=cm2</pre>
--	--

* 적용 시에 TB_SID를 tas, tiber0 중에 선택하게 하고 tas, tiber0 명령어로 TB_SID를 바꿀 수 있도록 구성

② .tas1_profile / .tas2_profile

<pre>export TB_SID=tas1 export TB_HOME=/home/tibero/tibero7 export TB_PROF_DIR=\$TB_HOME/bin/prof export PATH=\$TB_HOME/bin:\$TB_HOME/client/bin:\$TB_BASE/tbinary/ monitor:\$PATH export LD_LIBRARY_PATH=\$TB_HOME/lib:\$TB_HOME/client/lib:\$LD_LI BRARY_PATH</pre>	<pre>export TB_SID=tas2 export TB_HOME=/home/tibero/tibero7 export TB_PROF_DIR=\$TB_HOME/bin/prof export PATH=\$TB_HOME/bin:\$TB_HOME/client/bin:\$TB_BASE/tbinary/ monitor:\$PATH export LD_LIBRARY_PATH=\$TB_HOME/lib:\$TB_HOME/client/lib:\$LD_LI BRARY_PATH</pre>
---	---

* CM에서 TAS 인스턴스 등록 및 실행에 필요한 profile

③ .tiber01_profile / .tiber02_profile

<pre>export TB_SID=tiber01 export TB_HOME=/home/tibero/tibero7 export TB_PROF_DIR=\$TB_HOME/bin/prof export PATH=\$TB_HOME/bin:\$TB_HOME/client/bin:\$TB_BASE/tbinary/ monitor:\$PATH export LD_LIBRARY_PATH=\$TB_HOME/lib:\$TB_HOME/client/lib:\$LD_LI BRARY_PATH</pre>	<pre>export TB_SID=tiber02 export TB_HOME=/home/tibero/tibero7 export TB_PROF_DIR=\$TB_HOME/bin/prof export PATH=\$TB_HOME/bin:\$TB_HOME/client/bin:\$TB_BASE/tbinary/ monitor:\$PATH export LD_LIBRARY_PATH=\$TB_HOME/lib:\$TB_HOME/client/lib:\$LD_LI BRARY_PATH</pre>
--	--

* CM에서 TAC(Tibero) 인스턴스 등록 및 실행에 필요한 profile

(4) 바이너리 압축해제 및 라이선스 적용

- 양 서버에 바이너리 파일 압축을 해제합니다.

=> tar -xvzf tiber07-bin-FS02_PS04-linux64_4.18~tar.gz -C /home/tibero/

```
[tibero@server1:tas1:/home/tibero]ls -al
total 44
drwx----- 4 tibero dba 188 Nov 10 00:25 .
drwxr-xr-x 3 root root 20 Nov 9 23:39 ..
-rw----- 1 tibero dba 227 Nov 9 23:45 .bash_history
-rw-r--r-- 1 tibero dba 18 Apr 30 2024 .bash_logout
-rw-r--r-- 1 tibero dba 5628 Nov 10 00:20 .bash_profile
-rw-r--r-- 1 tibero dba 492 Nov 9 23:42 .bashrc
-rw-r--r-- 1 tibero dba 241 Nov 10 00:21 .tas1_profile
drwxr-x--- 12 tibero dba 145 Apr 30 2025 tbinary
-rw-r--r-- 1 tibero dba 244 Nov 10 00:22 .tiber01_profile
drwxr-x--- 9 tibero dba 97 Oct 15 09:20 tiber07
-rw----- 1 tibero dba 8223 Nov 10 00:22 .viminfo
-rw-r--r-- 1 tibero dba 19 Nov 9 23:44 .vimrc
[tibero@server1:tas1:/home/tibero]
```

```
[tibero@server2:tiber0:/home/tibero]ls -al
total 36
drwx----- 4 tibero dba 188 Nov 10 00:25 .
drwxr-xr-x 3 root root 20 Nov 9 23:39 ..
-rw----- 1 tibero dba 36 Nov 10 00:20 .bash_history
-rw-r--r-- 1 tibero dba 18 Apr 30 2024 .bash_logout
-rw-r--r-- 1 tibero dba 5644 Nov 10 00:20 .bash_profile
-rw-r--r-- 1 tibero dba 492 Apr 30 2024 .bashrc
-rw-r--r-- 1 tibero dba 241 Nov 10 00:23 .tas2_profile
drwxr-x--- 12 tibero dba 145 Apr 30 2025 tbinary
-rw-r--r-- 1 tibero dba 244 Nov 10 00:23 .tiber02_profile
drwxr-x--- 9 tibero dba 97 Oct 15 09:20 tiber07
-rw----- 1 tibero dba 3383 Nov 10 00:23 .viminfo
-rw-r--r-- 1 tibero dba 19 Nov 9 23:44 .vimrc
[tibero@server2:tiber0:/home/tibero]
```

- hostname과 ncpu에 맞는 데모라이선스를 발급하여 적용합니다.

=> cp license.xml \$TB_HOME/license/

```
[tibero@server1:tiber01:/home/tibero]cd $TB_HOME/license
[tibero@server1:tiber01:/home/tibero/tibero7/license]ls -arlt
total 4
drwxr-x--- 9 tibero dba 97 Oct 15 09:20 ..
drwxr-x--- 2 tibero dba 171 Oct 15 09:20 oss_licenses
-rw-r--r-- 1 tibero dba 576 Nov 10 00:25 license.xml
drwxr-x--- 3 tibero dba 45 Nov 10 00:25 .
[tibero@server1:tiber01:/home/tibero/tibero7/license]
```

```
[tibero@server2:tas2:/home/tibero]cd $TB_HOME/license
[tibero@server2:tas2:/home/tibero/tibero7/license]ls -arlt
total 4
drwxr-x--- 9 tibero dba 97 Oct 15 09:20 ..
drwxr-x--- 2 tibero dba 171 Oct 15 09:20 oss_licenses
-rw-r--r-- 1 tibero dba 576 Nov 10 00:25 license.xml
drwxr-x--- 3 tibero dba 45 Nov 10 00:25 .
[tibero@server2:tas2:/home/tibero/tibero7/license]
```

(5) 파라미터 파일 생성 및 설정 작업

- 파라미터 파일(tip 파일)은 DB용 파라미터 파일, TAS용 파라미터 파일, CM용 파라미터 파일까지 총 3개가 필요합니다. 추가로 tbsn.tbr 파일을 설정값에 맞게 수정해줍니다.

① DB용 파라미터 파일

- DB용 파라미터 파일을 생성하고 설정해줍니다.

=> export TB_SID=tibero1 (or tibero2) => sh \$TB_HOME/config/gen_tip.sh

```
[tibero@server1:tibero1:/home/tibero/tibero7/license]cd $TB_HOME/config
[tibero@server1:tibero1:/home/tibero/tibero7/config]sh gen_tip.sh
Using TB_SID "tibero1"
/home/tibero/tibero7/config/tibero1.tip generated
Already exists /home/tibero/tibero7/config/psm_commands!! Nothing has changed
/home/tibero/tibero7/client/config/tbsn.tbr generated.
Running client/config/gen_esql_cfg.sh
Done.
```

```
[tibero@server2:tibero2:/home/tibero/tibero7/license]cd $TB_HOME/config
[tibero@server2:tibero2:/home/tibero/tibero7/config]sh gen_tip.sh
Using TB_SID "tibero2"
/home/tibero/tibero7/config/tibero2.tip generated
Already exists /home/tibero/tibero7/config/psm_commands!! Nothing has changed
/home/tibero/tibero7/client/config/tbsn.tbr generated.
Running client/config/gen_esql_cfg.sh
Done.
```

=> vi \$TB_HOME/config/\$TB_SID.tip

<pre>##### ### Database Identification ##### DB_NAME=tibero1db LISTENER_PORT=8629 ##### ### File Configuration ##### CONTROL_FILES="+SYSTEM/control01.ctl", "+DATA/control02.ctl" DB_CREATE_FILE_DEST="+DATA" ##### ### TAS ##### USE_ACTIVE_STORAGE=Y AS_PORT=9629 ##### ### TAC ##### CLUSTER_DATABASE=Y THREAD=0 #1 UNDO_TABLESPACE=UNDO1 #UNDO2 LOCAL_CLUSTER_ADDR=192.168.182.128 #192.168.182.129 LOCAL_CLUSTER_PORT=11029 CM_PORT=11039 ##### ### Memory & Session ##### TOTAL_SHM_SIZE=2G MEMORY_TARGET=4G MAX_SESSION_COUNT=100 LOG_BUFFER=100M ##### ### Archive Log Mode ##### LOG_ARCHIVE_FORMAT="log-t%t-r%r-s%s.arc" LOG_ARCHIVE_DEST="+ARCH" _ARCHIVE_LAG_TARGET=7200 ##### ### TPR & Monitoring ##### TIBERO_PERFORMANCE_REPOSITORY=Y TPR_SNAPSHOT_SAMPLING_INTERVAL=60 TPR_SNAPSHOT_RETENTION=30 TPR_SNAPSHOT_TOP_SQL_CNT=10 TPR_SQL_PLAN_STAT=Y GATHER_SQL_EXEC_TIME=Y GATHER_SQL_PLAN_STAT=Y #BIND_VARIABLE_CAPTURE=Y #ACTIVE_SESSION_HISTORY=Y</pre>	<pre>##### ### Database Identification ##### DB_NAME=tibero2db LISTENER_PORT=8629 ##### ### File Configuration ##### CONTROL_FILES="+SYSTEM/control01.ctl", "+DATA/control02.ctl" DB_CREATE_FILE_DEST="+DATA" ##### ### TAS ##### USE_ACTIVE_STORAGE=Y AS_PORT=9629 ##### ### TAC ##### CLUSTER_DATABASE=Y THREAD=1 #0 UNDO_TABLESPACE=UNDO2 #UNDO1 LOCAL_CLUSTER_ADDR=192.168.182.129 #192.168.182.128 LOCAL_CLUSTER_PORT=11029 CM_PORT=11039 ##### ### Memory & Session ##### TOTAL_SHM_SIZE=2G MEMORY_TARGET=4G MAX_SESSION_COUNT=100 LOG_BUFFER=100M ##### ### Archive Log Mode ##### LOG_ARCHIVE_FORMAT="log-t%t-r%r-s%s.arc" LOG_ARCHIVE_DEST="+ARCH" _ARCHIVE_LAG_TARGET=7200 ##### ### TPR & Monitoring ##### TIBERO_PERFORMANCE_REPOSITORY=Y TPR_SNAPSHOT_SAMPLING_INTERVAL=60 TPR_SNAPSHOT_RETENTION=30 TPR_SNAPSHOT_TOP_SQL_CNT=10 TPR_SQL_PLAN_STAT=Y GATHER_SQL_EXEC_TIME=Y GATHER_SQL_PLAN_STAT=Y #BIND_VARIABLE_CAPTURE=Y #ACTIVE_SESSION_HISTORY=Y</pre>
--	--

<pre>##### # Performance ##### _LOG_WRITE_SYSCALL=1 _OPT_BIND_PEEKING=N _USE_DYNAMIC_SAMPLING=N _OPT_ADJUST_TABLE_PRED_SELECTIVITY=100 #Tibero7 _SGMT_SEARCH_METHOD=2 #Tibero7 _LOB_SGMT_SEARCH_METHOD=2 #Tibero7 _IDX_SGMT_SEARCH_METHOD=2 #Tibero7 ##### ## Miscellaneous ##### USE_TS_QUOTA=N USE_RECYCLEBIN=Y</pre>	<pre>##### # Performance ##### _LOG_WRITE_SYSCALL=1 _OPT_BIND_PEEKING=N _USE_DYNAMIC_SAMPLING=N _OPT_ADJUST_TABLE_PRED_SELECTIVITY=100 #Tibero7 _SGMT_SEARCH_METHOD=2 #Tibero7 _LOB_SGMT_SEARCH_METHOD=2 #Tibero7 _IDX_SGMT_SEARCH_METHOD=2 #Tibero7 ##### ## Miscellaneous ##### USE_TS_QUOTA=N USE_RECYCLEBIN=Y</pre>
--	--

* USE_ACTIVE_STORAGE: TAS 기능의 Active Storage를 사용하므로 Y (Default: N)

* AS_PORT: TAS 인스턴스의 LISTENER PORT

② TAS용 파라미터 파일

- TAS용 파라미터 파일을 생성하고 설정해줍니다.

=> export TB_SID=tas1 (or tas2) => sh \$TB_HOME/config/gen_tip.sh

<pre>[tibero@server1:tibero1:/home/tibero/tibero7/config]tas [tibero@server1:tas1:/home/tibero/tibero7/config]sh gen_tip.sh Using TB_SID "tas1" /home/tibero/tibero7/config/tas1.tip generated Already exists /home/tibero/tibero7/config/psm_commands!! Nothing has changed There's already /home/tibero/tibero7/client/config/tbdsn.tbr!! Added tas1 to /home/tibero/tibero7/client/config/tbdsn.tbr. Running client/config/gen_esql_cfg.sh Done. [tibero@server1:tas1:/home/tibero/tibero7/config]</pre>	<pre>[tibero@server2:tibero2:/home/tibero/tibero7/config]tas [tibero@server2:tas2:/home/tibero/tibero7/config]sh gen_tip.sh Using TB_SID "tas2" /home/tibero/tibero7/config/tas2.tip generated Already exists /home/tibero/tibero7/config/psm_commands!! Nothing has changed There's already /home/tibero/tibero7/client/config/tbdsn.tbr!! Added tas2 to /home/tibero/tibero7/client/config/tbdsn.tbr. Running client/config/gen_esql_cfg.sh Done. [tibero@server2:tas2:/home/tibero/tibero7/config]</pre>
---	---

=> vi \$TB_HOME/config/\$TB_SID.tip

<pre>##### ### Database Identification ##### DB_NAME=TAS LISTENER_PORT=9629 ##### ### TAS ##### INSTANCE_TYPE=AS AS_DISKSTRING="/dev/tas/*" AS_ALLOW_ONLY_RAW_DISKS=N ##### ### TAC ##### CLUSTER_DATABASE=Y THREAD=0 #1 LOCAL_CLUSTER_ADDR=192.168.182.128 #192.168.182.129 LOCAL_CLUSTER_PORT=11049 CM_PORT=11039 ##### ### Memory & Session ##### TOTAL_SHM_SIZE=1G MEMORY_TARGET=2G MAX_SESSION_COUNT=20 LOG_BUFFER=100M</pre>	<pre>##### ### Database Identification ##### DB_NAME=TAS LISTENER_PORT=9629 ##### ### TAS ##### INSTANCE_TYPE=AS AS_DISKSTRING="/dev/tas/*" AS_ALLOW_ONLY_RAW_DISKS=N ##### ### TAC ##### CLUSTER_DATABASE=Y THREAD=1 #0 LOCAL_CLUSTER_ADDR=192.168.182.129 #192.168.182.128 LOCAL_CLUSTER_PORT=11049 CM_PORT=11039 ##### ### Memory & Session ##### TOTAL_SHM_SIZE=1G MEMORY_TARGET=2G MAX_SESSION_COUNT=20 LOG_BUFFER=100M</pre>
---	---

* INSTANCE_TYPE: TAS 인스턴스의 파라미터 파일이므로 Y (Default: N)

* AS_DISKSTING: TAS 디스크를 탐색할 경로 지정

* AS_ALLOW_ONLY_RAW_DISKS: Raw Device가 아닌 Block Device에 디스크 링크를 이용하여 TAS 구성 예정으로 N (Default: Y)

③ CM용 파라미터 파일

- CM용 파라미터 파일을 생성하고 설정해줍니다.

=> **export CM_SID=cm1 (or cm2)**

=> **vi \$TB_HOME/config/\$CM_SID.tip**

CM_NAME=cm1 CM_UI_PORT=11039 CM_RESOURCE_FILE="/home/tibero/tibero7/config/cm1_res.c rf"	CM_NAME=cm2 CM_UI_PORT=11039 CM_RESOURCE_FILE="/home/tibero/tibero7/config/cm2_res.c rf"
---	---

④ tbdsn.tbr 파일

- tbdsn.tbr 파일을 수정해줍니다.

=> **vi \$TB_HOME/client/config/tbdsn.tbr**

tibero1=((INSTANCE=(HOST=localhost) (PORT=8629) (DB_NAME=tiberodb))) tas1=((INSTANCE=(HOST=localhost) (PORT=9629) (DB_NAME=TAS)))	tibero2=((INSTANCE=(HOST=localhost) (PORT=8629) (DB_NAME=tiberodb))) tas2=((INSTANCE=(HOST=localhost) (PORT=9629) (DB_NAME=TAS)))
--	--

* 파라미터 파일에 설정한 내용에 맞게 PORT, DB_NAME을 수정

3. TAS-TAC 설치

(1) 1번 노드 - TAS Diskspace 생성

- 1번 노드에서 TAS 인스턴스를 nomount로 기동하고 MASTER Diskspace를 생성합니다.

=> **export \$TB_SID=tas2**

=> **tboot nomount**

=> **Diskspace 생성**

```
[tibero@server1:tas1:/home/tibero]tas
[tibero@server1:tas1:/home/tibero]echo $TB_SID
tas1
[tibero@server1:tas1:/home/tibero]tboot nomount
Change core dump dir to /home/tibero/tibero7/bin/prof.
Listener port = 9629

Tibero 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Tibero instance started up (NOMOUNT mode).
[tibero@server1:tas1:/home/tibero]tbsql sys

tbSQL 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.

Enter Password:
Connected to Tibero.

SQL> CREATE DISKSPACE SYSTEM EXTERNAL REDUNDANCY
      2 DISK '/dev/tas/disk01' NAME disk01
      3 attribute 'AU_SIZE'='4M';

Diskspace 'SYSTEM' created.

SQL> q
Disconnected.
[tibero@server1:tas1:/home/tibero]tbproc
[tibero@server1:tas1:/home/tibero]
```

* Diskspace 이름 및 REDUNDANCY는 환경에 맞게 생성합니다.

(2) 1번 노드 - CM 설치 및 Diskspace, thread 추가

- 1번 노드에서 CM을 기동합니다.

=> **tbcm -b**

- 1번 서버 네트워크 리소스를 등록합니다.

=> **cmrctl add network --name tiberodb_inter1 --nettype private --ipaddr 192.168.182.128 --portno 11019**

=> **cmrctl add network --name tiberodb_pub1 --nettype public --ifname ens160**

=> **cmrctl show**

```
[tibero@server1:tas1:/home/tibero]tbcm -b
CM Guard daemon started up.

TBCM 7.1.1 (Build 303837)

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.

Tibero cluster manager started up.
Local node name is (cm1:11039).
```

```
[tiber@server1:tas1:/home/tibero]cmrctl add network --name tiberodb_inter1 --nettype private --ipaddr 192.168.182.128 --portno 11019
Resource add success! (network, tiberodb_inter1)
[tiber@server1:tas1:/home/tibero]cmrctl add network --name tiberodb_pub1 --nettype public --ifname ens160
Resource add success! (network, tiberodb_pub1)
[tiber@server1:tas1:/home/tibero]cmrctl show
Resource List of Node cml
=====
CLUSTER   TYPE      NAME      STATUS    DETAIL
-----
COMMON   network  tiberodb_inter1  UP (private) 192.168.182.128/11019
COMMON   network  tiberodb_pub1   UP (public)  ens160
=====
[tiber@server1:tas1:/home/tibero]
```

- 1번 서버 클러스터 리소스를 추가 후, 클러스터를 기동합니다.

=> **cmrctl add cluster --name tiberodb_cls --incnet tiberodb_inter1 --pubnet tiberodb_pub1 --cfile "+/dev/tas/*"**

=> **cmrctl start cluster --name tiberodb_cls * No root permission! 로그 무시**

=> **cmrctl show**

```
[tiber@server1:tas1:/home/tibero]cmrctl add cluster --name tiberodb_cls --incnet tiberodb_inter1 --pubnet tiberodb_pub1 --cfile "+/dev/tas/*"
Resource add success! (cluster, tiberodb_cls)
[tiber@server1:tas1:/home/tibero]cmrctl show
Resource List of Node cml
=====
CLUSTER   TYPE      NAME      STATUS    DETAIL
-----
COMMON   network  tiberodb_inter1  UP (private) 192.168.182.128/11019
COMMON   network  tiberodb_pub1   UP (public)  ens160
COMMON   cluster  tiberodb_cls    DOWN inc: tiberodb_inter1, pub: tiberodb_pub1
=====
[tiber@server1:tas1:/home/tibero]cmrctl start cluster --name tiberodb_cls
No root permission! vip is not available.
[tiber@server1:tas1:/home/tibero]cmrctl show
Resource List of Node cml
=====
CLUSTER   TYPE      NAME      STATUS    DETAIL
-----
COMMON   network  tiberodb_inter1  UP (private) 192.168.182.128/11019
COMMON   network  tiberodb_pub1   UP (public)  ens160
COMMON   cluster  tiberodb_cls    UP inc: tiberodb_inter1, pub: tiberodb_pub1
tiberodb_cls  file  tiberodb_cls:0  UP +0
tiberodb_cls  file  tiberodb_cls:1  UP +1
tiberodb_cls  file  tiberodb_cls:2  UP +2
=====
```

- 1번 서버 TAS 서비스와 인스턴스 리소스를 추가하고, TAS 인스턴스를 기동합니다.

=> **cmrctl add service --name TAS --cname tiberodb_cls --type as**

=> **cmrctl add as --name tas1 --svcname TAS --dbhome \$TB_HOME --envfile /home/tibero/.tas1_profile**

=> **cmrctl start as --name tas1**

```
[tiber@server1:tas1:/home/tibero]cmrctl add service --name tas --cname tiberodb_cls --type as
Resource add success! (service, tas)
[tiber@server1:tas1:/home/tibero]cmrctl add as --name tas1 --svcname tas --dbhome $TB_HOME --envfile /home/tibero/.tas1_profile
Resource add success! (as, tas1)

[tiber@server1:tas1:/home/tibero]cmrctl start as --name tas1
Change core dump dir to /home/tibero/tibero7/bin/prof.
Listener port = 9629

Tibero 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Tibero instance started up (NORMAL mode).
BOOT SUCCESS! (MODE : NORMAL)
```

- Diskspace와 쓰레드를 추가합니다.

=> **Diskspace 추가**

=> **alter diskpace SYSTEM add thread 1; * 마스터 디스크스페이스명으로 수행**

```

[tibero@server1:tas1:/home/tibero]tbsql sys
tbSQL 7
TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Enter Password:
Connected to Tibero.

SQL> CREATE DISKSPACE ARCH EXTERNAL REDUNDANCY
  2 DISK '/dev/tas/disk02' NAME disk02
  3 attribute 'AU_SIZE'='4M';

Diskspace 'ARCH' created.

SQL> CREATE DISKSPACE DATA EXTERNAL REDUNDANCY
  2 DISK '/dev/tas/disk03' NAME disk03
  3 DISK '/dev/tas/disk04' NAME disk04
  4 ATTRIBUTE 'AU_SIZE'='4M';

Diskspace 'DATA' created.

SQL> alter diskspace SYSTEM add thread 1;

Diskspace altered.

```

(3) 1번 노드 - DB 생성

- 1번 서버 DB 서비스와 인스턴스 리소스를 추가하고, DB 인스턴스를 nomount 기동합니다.
- => **cmrctl add service --name tiberodb --cname tiberodb_cls --type db=> cmrctl add as --name tas1 --svcname TAS --dbhome \$TB_HOME --envfile /home/tibero/.tas1_profile**
- => **cmrctl add db --name tiberol1 --svcname tiberodb --dbhome \$TB_HOME --envfile /home/tibero/.tiberol1_profile**

```

[tibero@server1:tiberol1:/home/tibero]echo $TB_SID
tiberol1
[tibero@server1:tiberol1:/home/tibero]cmrctl add service --name tiberodb --cname tiberodb_cls --type db
Resource add success! (service, tiberodb)
[tibero@server1:tiberol1:/home/tibero]cmrctl add db --name tiberol1 --svcname tiberodb --dbhome $TB_HOME --envfile /home/tibero/.tiberol1_profile
Resource add success! (db, tiberol1)

```

- DB 인스턴스를 NOMOUNT 기동 후, DB를 생성합니다.

=> **tbboot nomount**

```

[tibero@server1:tiberol1:/home/tibero]tbboot nomount
Change core dump dir to /home/tibero/tiberol1/bin/prof.
Listener port = 8629

Tibero 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Tibero instance started up (NOMOUNT mode).
[tibero@server1:tiberol1:/home/tibero]tbsql sys
tbSQL 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Enter Password:
Connected to Tibero.

```

```

SQL> create database
user sys identified by tiber0
character set MSWIN949
logfile thread 0 group 1 ('+SYSTEM/redo01_a.log', '+DATA/redo01_b.log') size 200M,
      thread 0 group 2 ('+SYSTEM/redo02_a.log', '+DATA/redo02_b.log') size 200M,
      thread 0 group 3 ('+SYSTEM/redo03_a.log', '+DATA/redo03_b.log') size 200M
maxdatafiles 1024
maxlogfiles 100
maxlogmembers 8
archivelog
  datafile '+SYSTEM/system_001.dtf' size 1G autoextend off
default tablespace USR
  datafile '+DATA/usr_001.dtf' size 3G autoextend off
default temporary tablespace TEMP
  tempfile '+DATA/temp_001.dtf' size 3G autoextend off
undo tablespace UNDO1
  datafile '+SYSTEM/undo1_001.dtf' size 3G autoextend off
SYSSUB
  datafile '+DATA/syssub_001.dtf' size 3G autoextend off
;

Database created.

```

- DB 기동 후, 2번 노드를 위한 UNDO, REDO를 추가하고 thread 1을 활성화합니다.

=> **tbboot**

```

[tibero@server1:tiberol:/home/tibero]tbboot
Change core dump dir to /home/tibero/tibero7/bin/prof.
Listener port = 8629

Tibero 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Tibero instance started up (NORMAL mode).
[tibero@server1:tiberol:/home/tibero]tbsql sys

tbSQL 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.

Enter Password:
Connected to Tibero.

SQL> create undo tablespace UNDO2 datafile '+SYSTEM/undo2_001.dtf' size 3G
autoextend off;

Tablespace 'UNDO2' created.

SQL> alter database add logfile thread 1 group 11 ('+SYSTEM/redo11_a.log', SQL>
'+DATA/redo11_b.log') size 200M;

Database altered.

SQL> alter database add logfile thread 1 group 12 ('+SYSTEM/redo12_a.log', SQL>
'+DATA/redo12_b.log') size 200M;

Database altered.

SQL> alter database add logfile thread 1 group 13 ('+SYSTEM/redo13_a.log',
'+DATA/redo13_b.log') size 200M;

Database altered.

SQL> alter database enable public thread 1;

Database altered.

```

(4) 1번 노드 - 시스템 오브젝트 생성

- 시스템 오브젝트를 생성합니다.

=> **sh system_install.sh -p1 tiber0 -p2 syscat**

```
[tiber0@server1:tiber01:/home/tiber0]tbhome
[tiber0@server1:tiber01:/home/tiber0/tiber07]cd scripts/
[tiber0@server1:tiber01:/home/tiber0/tiber07/scripts]sh system_install.sh -p1 tiber0 -p2 syscat
Creating internal system tables...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_dbms_apm.tbw...
Start TPR
Create tudi interface
Running /home/tiber0/tiber07/scripts/odci.sql...
Creating spatial meta tables and views ...
Registering default spatial reference systems ...
Registering unit of measure entries...
Creating internal system jobs...
Creating Japanese Lexer epa source ...
Creating internal system notice queue ...
Creating sql translator profiles ...
Creating agent table...
Creating label security packages ...
Running /home/tiber0/tiber07/scripts/label_security_system_user.sql...
Running /home/tiber0/tiber07/scripts/pkg/label_security_type.sql...
Running /home/tiber0/tiber07/scripts/systbl_label_security.sql...
Running /home/tiber0/tiber07/scripts/label_security_view.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_standard.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_lbac_standard.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_lbac_cache.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_lbac_rls.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_components.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_audit_admin.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_lbac_services.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_label_admin.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_lbac_sa_labels.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_util.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_policy_admin.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_session.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_sysdba.sql...
Running /home/tiber0/tiber07/scripts/pkg/pkg_sa_user_admin.sql...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_lbac_standard.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_lbac_cache.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_lbac_rls.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_components.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_audit_admin.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_lbac_services.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_label_admin.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_lbac_sa_labels.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_util.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_policy_admin.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_session.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_sysdba.tbw...
Running /home/tiber0/tiber07/scripts/pkg/_pkg_sa_user_admin.tbw...
Running /home/tiber0/tiber07/scripts/label_security_packages.sql...
Running /home/tiber0/tiber07/scripts/label_security_synonym.sql...
Setting system_install.sh parameter...
Done.
For details, check /home/tiber0/tiber07/instance/tiber01/log/system_init.log.
```

(5) 2번 노드 - CM 설치

- 2번 노드에서 CM을 기동합니다.

=> **tbcm -b**

- 2번 서버 네트워크 리소스를 등록합니다.

=> **cmrctl add network --name tiberodb_inter2 --nettype private --ipaddr 192.168.182.129 --portno 11019**

=> **cmrctl add network --name tiberodb_pub2 --nettype public --ifname ens160**

=> **cmrctl show**

```
[tiber@server2:tibero2:/home/tibero]tbcm -b
CM Guard daemon started up.

TBCM 7.1.1 (Build 303837)

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.

Tibero cluster manager started up.
Local node name is (cm2:11039).

[tiber@server2:tibero2:/home/tibero]cmrctl add network --name tiberodb_inter2 --nettype private --ipaddr 192.168.182.129 --portno 11019
Resource add success! (network, tiberodb_inter2)
[tiber@server2:tibero2:/home/tibero]cmrctl add network --name tiberodb_pub2 --nettype public --ifname ens160
Resource add success! (network, tiberodb_pub2)
[tiber@server2:tibero2:/home/tibero]cmrctl show
Resource List of Node cm2
=====
CLUSTER   TYPE      NAME      STATUS    DETAIL
-----
COMMON   network  tiberodb_inter2  UP (private) 192.168.182.129/11019
COMMON   network  tiberodb_pub2    UP (public) ens160
=====
[tiber@server2:tibero2:/home/tibero]
```

- 2번 서버 클러스터 리소스를 추가 후, 클러스터를 기동합니다.

=> **cmrctl add cluster --name tiberodb_cls --incnet tiberodb_inter2 --pubnet tiberodb_pub2 --cfile "+/dev/tas/*"**

=> **cmrctl start cluster --name tiberodb_cls * No root permission! 로그 무시**

=> **cmrctl show**

```
[tiber@server2:tibero2:/home/tibero]cmrctl add cluster --name tiberodb_cls --incnet tiberodb_inter2 --pubnet tiberodb_pub2 --cfile "+/dev/tas/*"
Resource add success! (cluster, tiberodb_cls)
[tiber@server2:tibero2:/home/tibero]cmrctl show
Resource List of Node cm2
=====
CLUSTER   TYPE      NAME      STATUS    DETAIL
-----
COMMON   network  tiberodb_inter2  UP (private) 192.168.182.129/11019
COMMON   network  tiberodb_pub2    UP (public) ens160
COMMON   cluster  tiberodb_cls     DOWN inc: tiberodb_inter2, pub: tiberodb_pub2
=====
[tiber@server2:tibero2:/home/tibero]cmrctl start cluster --name tiberodb_cls
No root permission! vip is not available.
[tiber@server2:tibero2:/home/tibero]cmrctl show
Resource List of Node cm2
=====
CLUSTER   TYPE      NAME      STATUS    DETAIL
-----
COMMON   network  tiberodb_inter2  UP (private) 192.168.182.129/11019
COMMON   network  tiberodb_pub2    UP (public) ens160
COMMON   cluster  tiberodb_cls     UP inc: tiberodb_inter2, pub: tiberodb_pub2
tiberodb_cls  file  tiberodb_cls:0  UP +0
tiberodb_cls  file  tiberodb_cls:1  UP +1
tiberodb_cls  file  tiberodb_cls:2  UP +2
tiberodb_cls  service  tas             UP Active Storage, Active Cluster (auto-restart: OFF)
tiberodb_cls  service  tiberodb       UP Database, Active Cluster (auto-restart: OFF)
=====
[tiber@server2:tibero2:/home/tibero]
```

(6) 2번 노드 - TAS 인스턴스 설치

- 2번 서버 TAS 인스턴스 리소스를 추가하고, TAS 인스턴스를 기동합니다.

=> **cmrctl add as --name tas2 --svcname TAS --dbhome \$TB_HOME --envfile /home/tibero/.tas2_profile**

=> **cmrctl start as --name tas2**

```
[tiber@server2:tibero2:/home/tibero]cmrctl add as --name tas2 --svcname tas --dbhome $TB_HOME --envfile /home/tibero/.tas2_profile
Resource add success! (as, tas2)
[tiber@server2:tibero2:/home/tibero]cmrctl start as --name tas2
Change core dump dir to /home/tibero/tibero7/bin/prof.
Listener port = 9629

Tibero 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Tibero instance started up (NORMAL mode).
BOOT SUCCESS! (MODE : NORMAL)
```

(7) 2번 노드 - DB 인스턴스 설치

- 2번 서버 DB 인스턴스 리소스를 추가하고, DB 인스턴스를 기동합니다.

```
=> cmrctl add db --name tibero2 --svcname tiberodb --dbhome $TB_HOME --envfile /home/tibero/.tibero2_profile
```

```
=> cmrctl start db --name tibero2
```

```
=> cmrctl show
```

```
[tibero@server2:tibero2:/home/tibero]cmrctl add db --name tibero2 --svcname tiberodb --dbhome $TB_HOME --envfile /home/tibero/.tibero2_profile
Resource add success! (db, tibero2)
[tibero@server2:tibero2:/home/tibero]cmrctl start db --name tibero2
Change core dump dir to /home/tibero/tibero7/bin/prof.
Listener port = 8629

Tibero 7

TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Tibero instance started up (NORMAL mode).
BOOT SUCCESS! (MODE : NORMAL)
[tibero@server2:tibero2:/home/tibero]cmrctl show
Resource List of Node cm2
=====
CLUSTER   TYPE      NAME      STATUS      DETAIL
-----
COMMON   network  tiberodb_inter2  UP (private) 192.168.182.129/11019
COMMON   network  tiberodb_pub2    UP (public) ens160
COMMON   cluster  tiberodb_cls     UP inc: tiberodb_inter2, pub: tiberodb_pub2
tiberodb_cls  file  tiberodb_cls:0    UP +0
tiberodb_cls  file  tiberodb_cls:1    UP +1
tiberodb_cls  file  tiberodb_cls:2    UP +2
tiberodb_cls  service tas               UP Active Storage, Active Cluster (auto-restart: OFF)
tiberodb_cls  service tiberodb         UP Database, Active Cluster (auto-restart: OFF)
tiberodb_cls  as      tas2 UP(NRML) tas, /home/tibero/tibero7, failed retry cnt: 0
tiberodb_cls  db      tibero2 UP(NRML) tiberodb, /home/tibero/tibero7, failed retry cnt: 0
=====
[tibero@server2:tibero2:/home/tibero]
```

(8) 설치 및 동기화 확인

- 1번 노드 설치 확인 (Status에 NRML)

```
[tibero@server1:tiberol:/home/tibero]cmrctl show
Resource List of Node cm1
=====
CLUSTER   TYPE      NAME      STATUS      DETAIL
-----
COMMON   network  tiberodb_inter1  UP (private) 192.168.182.128/11019
COMMON   network  tiberodb_pub1    UP (public) ens160
COMMON   cluster  tiberodb_cls     UP inc: tiberodb_inter1, pub: tiberodb_pub1
tiberodb_cls  file  tiberodb_cls:0    UP +0
tiberodb_cls  file  tiberodb_cls:1    UP +1
tiberodb_cls  file  tiberodb_cls:2    UP +2
tiberodb_cls  service tas               UP Active Storage, Active Cluster (auto-restart: OFF)
tiberodb_cls  service tiberodb         UP Database, Active Cluster (auto-restart: OFF)
tiberodb_cls  as      tas1 UP(NRML) tas, /home/tibero/tibero7, failed retry cnt: 0
tiberodb_cls  db      tiberol UP(NRML) tiberodb, /home/tibero/tibero7, failed retry cnt: 0
=====
[tibero@server1:tiberol:/home/tibero]
```

- 2번 노드 설치 확인 (Status에 NRML)

```
[tibero@server2:tibero2:/home/tibero]cmrctl show
Resource List of Node cm2
=====
CLUSTER   TYPE      NAME      STATUS      DETAIL
-----
COMMON   network  tiberodb_inter2  UP (private) 192.168.182.129/11019
COMMON   network  tiberodb_pub2    UP (public) ens160
COMMON   cluster  tiberodb_cls     UP inc: tiberodb_inter2, pub: tiberodb_pub2
tiberodb_cls  file  tiberodb_cls:0    UP +0
tiberodb_cls  file  tiberodb_cls:1    UP +1
tiberodb_cls  file  tiberodb_cls:2    UP +2
tiberodb_cls  service tas               UP Active Storage, Active Cluster (auto-restart: OFF)
tiberodb_cls  service tiberodb         UP Database, Active Cluster (auto-restart: OFF)
tiberodb_cls  as      tas2 UP(NRML) tas, /home/tibero/tibero7, failed retry cnt: 0
tiberodb_cls  db      tibero2 UP(NRML) tiberodb, /home/tibero/tibero7, failed retry cnt: 0
=====
[tibero@server2:tibero2:/home/tibero]
```

- 동기화 확인

=> 1번 서버에서 테스트 테이블 생성 후 데이터 삽입

=> 2번 서버에서 테스트 데이터 정상 조회 확인

```
[tibero@server1:tibero1:/home/tibero]tbsql tibero
tbSQL 7
TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Enter Password:
Connected to Tibero.
SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
tibero1

1 row selected.
SQL> create table test (c1 number);
Table 'TEST' created.
SQL> insert into test values (999999);
1 row inserted.
SQL> commit;
Commit completed.

[tibero@server2:tibero2:/home/tibero]tbsql tibero
tbSQL 7
TmaxTibero Corporation Copyright (c) 2020-. All rights reserved.
Enter Password:
Connected to Tibero.
SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
tibero2

1 row selected.
SQL> select * from test;

          C1
-----
          999999

1 row selected.
SQL>
SQL>
SQL>
```