

OLVM 설정 최적화

OLVM 설치 및 최적화 가이드

아래는 OLVN 설치 직후부터 운영 단계까지 성능 최적화를 염두에 두고 설정해야 할 주요 항목들을 단계별로 정리한 가이드입니다.

1. 베이스 OS 준비 단계

1. 운영체제 선택

- 권장: Oracle Linux 8.10 또는 RHEL/CentOS 8.x
- 최신 커널(5.x 계열) 적용으로 가상화 성능·안정성↑

2. 커널 파라미터 튜닝 (/etc/sysctl.conf 예시)

```
vm.swappiness = 10
vm.dirty_ratio = 15
vm.dirty_background_ratio = 5
fs.file-max = 2097152
net.core.somaxconn = 4096
net.ipv4.tcp_fin_timeout = 15
```

적용:

```
sysctl -p
```

3. HugePages 활성화

- 메모리 단편화를 줄이고 TLB 미스 감소
- /etc/default/grub 에서 GRUB_CMDLINE_LINUX에 추가:

```
default_hugepagesz=1G hugepagesz=1G hugepages=4
```

- grub2-mkconfig -o /boot/grub2/grub.cfg 후 재부팅

4. NUMA 바인딩(선택)

- CPU와 메모리를 노드 단위로 묶어 성능 편차 최소화
- numactl 또는 VM 단위로 pCPU pinning 설정

5. 시간 동기화

```
dnf install chrony
systemctl enable --now chronyd
chronyc sources
```

- OLVM 엔진·호스트 간 시계 불일치는 클러스터 장애 유발

2. OLVM Engine(Manager) 설치 및 튜닝

1. 엔진 패키지 설치

```
dnf install ovirt-engine
engine-setup
```

- DB 암호, 네트워크, 스토리지 리포지토리 지정

2. PostgreSQL 튜닝 (`/var/lib/pgsql/data/postgresql.conf` 예시)

```
shared_buffers = 4GB
effective_cache_size = 12GB
maintenance_work_mem = 1GB
checkpoint_completion_target = 0.9
wal_buffers = 16MB
max_wal_size = 4GB
```

변경 후: `systemctl restart postgresql`

3. Java JVM 옵션 조정 (`/etc/ovirt-engine/jvm.conf`)

```
Xms=4g
Xmx=8g
XX:MetaspaceSize=512m
XX:MaxMetaspaceSize=1g
```

4. 로그·이벤트 보존 주기

- `/etc/ovirt-engine/engine.conf.d/99-log-rotation.conf`

```
AuditLogMaxDays=30
EventLogMaxDays=15
```

3. 호스트(Compute Node) 설정

1. ovirt-host 패키지 설치

```
dnf install ovirt-node
```

2. Tuned 프로파일 사용

```
dnf install tuned
tuned-adm profile virtualization-host
```

3. CPU 핀닝 & NUMA

- VM별 vCPU를 특정 pCPU에 고정(pin)

4. HugePages 적용

- `/etc/libvirt/qemu.conf`:

```
memoryBackingHugepagesSize = "1G"
memoryBackingHugepages = true
```

4. 스토리지 설정

1. 스토리지 도메인 선택

- Raw LVM: 성능 최상
- QCOW2 (cache=none, io=native): 스냅샷 기능 활용

2. i/o 스케줄러

```
echo noop > /sys/block/sdX/queue/scheduler
```

3. VM 디스크 캐시 모드

- Host-cache=false, Write-Through 또는 Direct I/O 권장
- “Enable Virtio SCSI” & “IO Mode: native” 설정

4. 분산 스토리지(Ceph/GlusterFS)

- `rbd cache` 활성화
- `rbd_default_features = 3`

5. 네트워크 설정

1. NIC 본딩 & VLAN 분리

- `bond0`: 관리/engine 통신
- `bond1`: VM 데이터
- `bond2`: 스토리지(iSCSI/NFS)
- `bond3`: 라이브마이그레이션

2. MTU 설정 (최소 9000)

```
ip link set dev bond0 mtu 9000
```

3. SR-IOV / VFIO (선택)

4. VLAN 태깅 (UI → Networks → VLAN ID 지정)

6. 추가 최적화 & 운영 팁

- 메모리 오버커밋 제한: Memory Overcommit slider 조정

- **CPU 토폴로지:** vCPU sockets/cores/threads 최적화
- **Host affinity/anti-affinity:** VM 충돌 방지
- **Cache 모니터링:** `virt-top`, `iostat`, `htop`
- **업데이트 관리:** 호스트 롤링 재부팅
- **백업/스냅샷:** `ovirt-engine-backup`
- **모니터링:** Prometheus + Grafana (`ovirt_exporter`)

Note: 하드웨어 스펙, 워크로드 등에 따라 벤치마크(`fio`, `iperf3`, `virt-top`)로 효과 확인 필수.