목차

Openstack CLI Command 3
Identity (Keystone) 3
Images (glance) 3
Compute (Nova) 4
Networking (Neutron) 6
Block Storage (Cinder) 6
Object Storage (Swift) 6

Openstack CLI Command

Identity (Keystone)

```
# 사용자 목록
$ keystone user-list

# ID서비스 카탈로그
$ keystone catalog

# keystone endpoint 탐색
$ keystone discover

# 서비스 카탈로그 전체 조회
$ keystone service-list

# 새로운 사용자 생성
$ keystone user-create --name --tenant-id --pass --email --enabled

#새로운 테넌트 생성
$ keystone tenant-create --name --description --enabled
```

Images (glance)

```
# 접근 가능한 이미지 목록
$ glance image-list

# 특정 이미지 삭제
$ glance image-delete <image>

# 이미지 조회
$ glance image-show <image>

# 이미지 업데이트
$ glance image-update <image>

# 커널 이미지 업로드
$ glance image-create --name "cirros-threepart-kernel" --disk-format aki --
container-format aki --is-public False --file ~/images/cirros-0.3.1~pre4-
x86_64-vmlinuz

# RAM 이미지 업로드
$ glance image-create --name "cirros-threepart-ramdisk" --disk-format ari --
container-format ari --is-public False --file ~/images/cirros-0.3.1~pre4-
x86_64-initrd
```

```
# three-part 이미지 업로드
$ glance image-create --name "cirros-threepart" --disk-format ami --
container-format ami --is-public False --property kernel_id=$KID-property
ramdisk_id=$RID --file ~/images/cirros-0.3.1~pre4-x86_64-blank.img

# RAW 이미지 등록
$ glance image-create --name "cirros-qcow2" --disk-format qcow2 --container-
format bare --is-public False --file ~/images/cirros-0.3.1~pre4-x86_64-
disk.img
```

Compute (Nova)

```
# 인스턴스 체크 및 목록
$ nova list

# 이미지 목록
$ nova image-list

# flavor 목록
$ nova flavor-list

# flavor와 이미지명으로 인스턴스 생성
$ nova boot --image <image> --flavor <flavor_instance_name>
$ nova boot --image cirros-0.3.1-x86_64-uec --flavor m1.tiny MyFirstInstance

# 인스턴스 로그인
$ ip netns
$ ip netns exec ssh <user@server or use a key>
$ ip netns exec qdhcp-6021a3b4-8587-4f9c-8064-0103885dfba2 ssh
cirros@10.0.0.2

# 인스턴스 상세정보
$ nova show <name>
$ nova show <id>

# 인스턴스 콘솔 로그
$ nova console-log <name>

# 인스턴스 메타정보 설정
$ nova meta <image_name> set newmeta='my meta data'

# 인스턴스 스냅샷 생성
$ nova image-create volumeTwoImage snapshotOfVolumeImage
$ nova image-show snapshotOfVolumeImage

# 일시정지 (pause)
$ nova pause <name>
$ nova pause volumeTwoImage

# 재시작
```

```
$ nova unpause <name>

# 멈춤 (Suspend)
$ nova suspend <name>

# 정지 (Stop)
$ nova stop <name>

# 기동 (Start)
$ nova start <name>

# 응급복구 (Rescue)
$ nova rescue <name>
$ nova rescue <name --rescue_image_ref <rescue_image>

# 리사이즈 (Resize)
$ nova resize <name> <flavor>
$ nova resize my-pem-server m1.small
$ nova resize-confirm my-pem-server1

# 리빌드 (Rebuild)
$ nova rebuild <name> <image>
$ nova rebuild newtinny cirros-qcow2

# 리부트 (Reboot)
$ nova reboot <name>
$ nova reboot newtinny

# 인스턴스에 사용자데이터와 파일 전송
$ nova boot --user-data <file> <instance>
$ nova boot --user-data userdata.txt --image cirros-qcow2 --flavor m1.tiny
MyUserDataInstance2

# 키페어 생성
$ nova keypair-add test > test.pem
$ chmod 600 test.pem

# 인스턴스 시작 (Boot)
$ nova boot --image cirros-0.3.0-x86_64 --flavor m1.small --key_name test
MyFirstServer

# 인스턴스에 SSH 접속
$ ip netns exec qdhcp-98f09f1e-64c4-4301-a897-5067ee6d544f ssh -i test.pem
cirros@10.0.0.4

# 보안그룹 관리
$ nova secgroup-add-group-rule default default icmp -1 -1
$ nova secgroup-add-group-rule default default tcp 22 22
```

Networking (Neutron)

```
# 네트워크 생성
$ neutron net-create <name>

# 서브넷 생성
$ neutron subnet-create NETWORK_NAME CIDR
$ neutron subnet-create my-network 10.0.0.0/29

# 네트워크, 서브넷 조회
$ neutron net-list
$ neutron subnet-list
```

Block Storage (Cinder)

```
# 새 볼륨 생성
$ cinder create <size_in_GB> --display-name <NAME>
$ cinder create 10 --display-name MyFirstVolume

# 볼륨 추가 후 인스턴스 부팅
$ nova boot --image cirros-qcow2 --flavor m1.tiny MyVolumeInstance

# 볼륨 목록
$ cinder list

# 기동 중인 인스턴스에 볼륨 추가
$ nova volume-attach <INSTANCE_ID> <VOLUME_ID> auto
$ nova volume-attach MyVolumeInstance /dev/vdb auto
```

Object Storage (Swift)

```
# 계정, 컨테이너, 오브젝트 정보 조회
$ swift stat
$ swift stat <ACCOUNT>
$ swift stat <CONTAINER>
$ swift stat <OBJECT>

# 컨테이너 목록
$ swift list

# 컨테이너 생성
$ swift post mycontainer

# 컨테이너에 파일 업로드
$ swift upload <container_name> <file_name>
$ swift upload mycontainer myfile.txt
```

```
# 컨테이너 내 오브젝트 목록
$ swift list container

# 컨테이너에서 오브젝트 다운로드
$ swift download <container_name> <object_name>
```

From:
https://atl.kr/dokuwiki/ - **AllThatLinux!**

Permanent link:
https://atl.kr/dokuwiki/doku.php/openstack_cli_command

Last update: **2020/06/12 07:52**