LXC (리눅스 컨테이너)

LXC는 리눅스 커널 컨테이너 기능을 위한 사용자영역 인터페이스입니다. 강력한 API와 간단한 도구들을 통해 리눅스 사용자가 쉽게 시스템 또는 어플리케이션 컨테이너들을 생성/관리할 수 있게 해줍니다.

현재 LXC는 컨테이너를 만들기 위해 아래의 커널 기능들을 사용하고 있습니다.

•커널 네임스페이스 (ipc, uts, 마운트, pid, 네트워크, 사용자)

•Apparmor와 SELinux 프로파일

•Seccomp 정책

•Chroots (pivot\_root 사용)

•커널 캐퍼빌리티

•CGroups (컨트롤 그룹)

[LXC 설치]

1. lxc 데몬 설치

**root@ubuntu14:~# apt-get install lxc**

2 . 설치확인

root@ubuntu14:~# **lxc-checkconfig**

Kernel configuration not found at /proc/config.gz; searching...

Kernel configuration found at /boot/config-4.2.0-27-generic

--- Namespaces ---

Namespaces: enabled

Utsname namespace: enabled

Ipc namespace: enabled

Pid namespace: enabled

User namespace: enabled

Network namespace: enabled

Multiple /dev/pts instances: enabled

--- Control groups ---

Cgroup: enabled

Cgroup clone\_children flag: enabled

Cgroup device: enabled

Cgroup sched: enabled

Cgroup cpu account: enabled

Cgroup memory controller: enabled

Cgroup cpuset: enabled

--- Misc ---

Veth pair device: enabled

Macvlan: enabled

Vlan: enabled

Bridges: enabled

Advanced netfilter: enabled

CONFIG\_NF\_NAT\_IPV4: enabled

CONFIG\_NF\_NAT\_IPV6: enabled

CONFIG\_IP\_NF\_TARGET\_MASQUERADE: enabled

CONFIG\_IP6\_NF\_TARGET\_MASQUERADE: enabled

CONFIG\_NETFILTER\_XT\_TARGET\_CHECKSUM: enabled

--- Checkpoint/Restore ---

checkpoint restore: enabled

CONFIG\_FHANDLE: enabled

CONFIG\_EVENTFD: enabled

CONFIG\_EPOLL: enabled

CONFIG\_UNIX\_DIAG: enabled

CONFIG\_INET\_DIAG: enabled

CONFIG\_PACKET\_DIAG: enabled

CONFIG\_NETLINK\_DIAG: enabled

File capabilities: enabled

Note : Before booting a new kernel, you can check its configuration

usage : CONFIG=/path/to/config /usr/bin/lxc-checkconfig정확인

3. 컨테이너 생성

root@ubuntu14:~# **lxc-create -n test -t ubuntu**

Checking cache download in /var/cache/lxc/trusty/rootfs-amd64 ...

Installing packages in template: ssh,vim,language-pack-en

Downloading ubuntu trusty minimal ...

I: Retrieving Release

I: Retrieving Release.gpg

4. 컨테이너 상태

root@ubuntu14:~# **lxc-ls --fancy**

NAME STATE IPV4 IPV6 AUTOSTART

------------------------------------

test STOPPED - - NO

5. 컨테이너 실행

root@ubuntu14:~# **lxc-start -n test -d**

root@ubuntu14:~# **lxc-ls --fancy**

NAME STATE IPV4 IPV6 AUTOSTART

------------------------------------------

test RUNNING 10.0.3.126 - NO

root@ubuntu14:~# **lxc-info --name test**

Name: test

State: RUNNING

PID: 36568

IP: 10.0.3.126

CPU use: 1.09 seconds

BlkIO use: 992.00 KiB

Memory use: 10.70 MiB

KMem use: 0 bytes

Link: vethT439FP

TX bytes: 3.45 KiB

RX bytes: 3.65 KiB

Total bytes: 7.10 KiB

6. 컨테이너 접속

root@ubuntu14:~# **lxc-console -n test**

Ubuntu 14.04.5 LTS test tty1

test login: **ubuntu**

Password: **ubuntu**

Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.2.0-27-generic x86\_64)

\* Documentation: https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;

the exact distribution terms for each program are described in the

individual files in /usr/share/doc/\*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by

applicable law.

ubuntu@test:~$ ip a

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid\_lft forever preferred\_lft forever

inet6 ::1/128 scope host

valid\_lft forever preferred\_lft forever

5: eth0@if6: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP group default qlen 1000

link/ether 00:16:3e:fb:73:dc brd ff:ff:ff:ff:ff:ff

inet 10.0.3.126/24 brd 10.0.3.255 scope global eth0

valid\_lft forever preferred\_lft forever

inet6 fe80::216:3eff:fefb:73dc/64 scope link

valid\_lft forever preferred\_lft forever

ubuntu@test:~$ **sudo apt-get install nginx**

ubuntu@test:~$ **sudo service nginx start**

7. 서버에서 포트포워딩 설정(물리서버의 9090포트-> 컨테이너 80으로 포워딩)

root@ubuntu14:/# **iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 9090 -j DNAT --to 10.0.3.126:80**

웹서버 접속확인 (http://물리서버의ip:9090)

8. 컨테이너 설정

/var/lib/lxc/test 디렉토리의 config 파일에서 관리

root@ubuntu14:/var/lib/lxc/test# cat config

# Template used to create this container: /usr/share/lxc/templates/lxc-ubuntu

# Parameters passed to the template:

# For additional config options, please look at lxc.container.conf(5)

# Common configuration

lxc.include = /usr/share/lxc/config/ubuntu.common.conf

# Container specific configuration

lxc.rootfs = /var/lib/lxc/test/rootfs

lxc.mount = /var/lib/lxc/test/fstab

lxc.utsname = test

lxc.arch = amd64

# Network configuration

lxc.network.type = veth

lxc.network.flags = up

lxc.network.link = lxcbr0

lxc.network.hwaddr = 00:16:3e:fb:73:dc

메모리 50M제한설정: lxc.cgroup.memory.limit\_in\_bytes = 50000000

cpu제한설정 : lxc.cgroup.cpu.shares = 100

7. 컨테이너 중지

root@ubuntu14:~# **lxc-stop -n test**

root@ubuntu14:~# **lxc-ls --fancy**

NAME STATE IPV4 IPV6 AUTOSTART

------------------------------------

test STOPPED - - NO

8. 컨테이너 삭제

root@ubuntu14:~# **lxc-destroy -n test**

root@ubuntu14:~# **lxc-ls --fancy**

NAME STATE IPV4 IPV6 AUTOSTART