목차

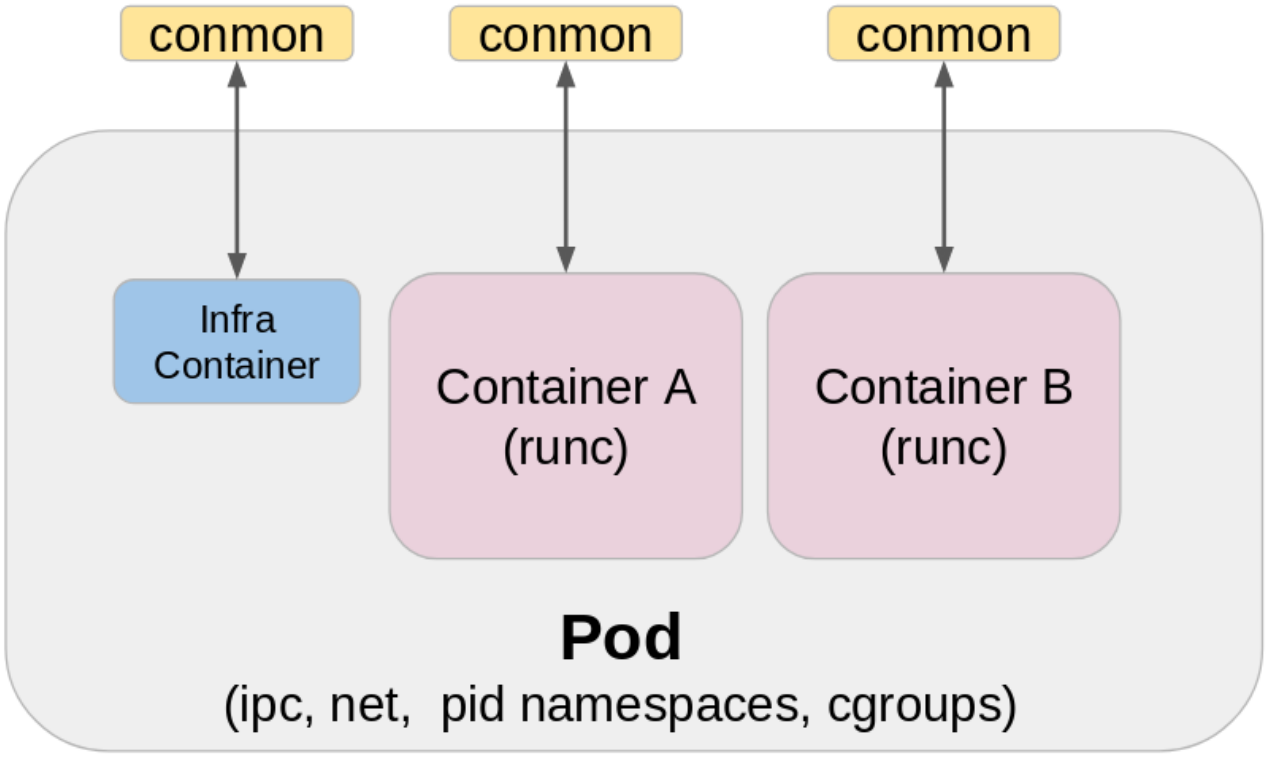
Podman 로컬 컨테이너 런타임에서 포드 및 컨테이너 관리 3
Podman Pods: 알아야 할 사항 3
The CLI: podman pod 4
Create a pod 4
Add a container to a pod 6
Shortcut to create pods 6
Pods and container management 7
참조링크 9

Last update: 2021/07/27 01:57	podman_로컬_컨테이너_런타임에서_포드_및_컨테이너_관리	https://atl.kr/dokuwiki/doku.php/podman_%EB%A1%9C%EC%BB%AC_%EC%BB%A8%ED%85%8C%EC%9D%B4%EB%84%88_%EB%9F%B0%ED%83%80%EC%9E%84%EC%97%90%EC%84%9C_%ED%8F%AC%EB%93%9C_%EB%B0%8F_%EC%BB%A8%ED%85%8C%EC%9D%B4%EB%84%88_%EA%B4%80%EB%A6%AC

Podman 로컬 컨테이너 런타임에서 포드 및 컨테이너 관리

Podman Pods: 알아야 할 사항

쿠버네티스에서 POD 개념이 도입되었습니다. Podman POD는 쿠버네티스와 유사합니다.



모든 Podman 포드에는 인프라 컨테이너가 포함되어 있습니다. 이 컨테이너는 아무 작업도 수행하지 않지만 잠자기 상태로 전환됩니다. 그 목적은 포드와 연결된 네임스페이스를 보유하고 포드맨이 다른 컨테이너를 포드에 연결할 수 있도록 하는 것입니다. 이를 통해 POD 내에서 컨테이너를 시작 및 중지할 수 있으며 포드는 계속 실행됩니다. 기본 컨테이너가 포드를 제어하는 것처럼 이는 불가능합니다. 기본 인프라 컨테이너는 k8s.gcr.io/pause이미지를 기반으로 합니다 . 달리 명시하지 않는 한 모든 포드에는 기본 이미지 기반 컨테이너가 있습니다.

Pod를 구성하는 대부분의 속성은 실제로 infra 컨테이너에 할당됩니다. 포트 바인딩, cgroup-parent 값 및 커널 네임스페이스는 모두 infra 컨테이너에 할당됩니다. 포드가 생성되면 이러한 속성이 인프라 컨테이너에 할당되고 변경할 수 없기 때문에 이를 이해하는 것이 중요합니다. 예를 들어 포드를 생성한 다음 나중에 새 포트를 바인딩하는 컨테이너를 추가하기로 결정하면 포드맨은 이를 수행할 수 없습니다. 새 컨테이너를 추가하기 전에 추가 포트 바인딩으로 포드를 다시 생성해야 합니다.

위의 다이어그램에서 각 컨테이너 위에 있는 상자를 확인하십시오. 이것은 컨테이너 모니터(conmon)입니다. 작은 C 프로그램이 하는 일은 컨테이너의 기본 프로세스를 감시하고 컨테이너가 죽으면 종료 코드를 저장하는 것입니다. 또한 나중에 첨부할 수 있도록 컨테이너의 tty를 열어둡니다. 이것은 podman이 분리 모드(백그라운드)에서 실행되도록 하여 podman은 종료할 수 있지만 conmon은 계속 실행됩니다. 각 컨테이너에는 고유한 conmon 인스턴스가 있습니다.

The CLI: podman pod

우리는 `podman pod`명령을 통해 파드와의 상호작용의 대부분을 노출 합니다. 다른 작업 중에서 `podman pod`포드를 생성, 삭제, 쿼리 및 검사하는 데 사용할 수 있습니다. `podman pod`인수 없이 실행하면 모든 포드 관련 명령을 볼 수 있습니다.

```
$ sudo podman pod

NAME:
   podman pod - Manage container pods.


Pods are a group of one or more containers sharing the same network, pid and
ipc namespaces.

USAGE:
   podman pod command [command options] [arguments...]


COMMANDS:
    create        Create a new empty pod
    exists        Check if a pod exists in local storage
    inspect       displays a pod configuration
    kill          Send the specified signal or SIGKILL to containers in pod
    pause         Pause one or more pods
    ps, ls, list  List pods
    restart       Restart one or more pods
    rm            Remove one or more pods
    start         Start one or more pods
    stats         Display percentage of CPU, memory, network I/O, block I/O
and PIDs for containers in one or more pods
    stop          Stop one or more pods
    top           Display the running processes of containers in a pod
    unpause       Unpause one or more pods

OPTIONS:
   --help, -h  show help
```

Create a pod

Podman으로 포드를 만드는 전통적인 방법은 `podman pod create`명령을 사용하는 것 입니다.

```
$ sudo podman pod create --help

NAME:
```

```
   podman pod create - Create a new empty pod

USAGE:
   podman pod create [command options] [arguments...]

DESCRIPTION:
   Creates a new empty pod. The pod ID is then printed to stdout. You can
then start it at any time with the podman pod start <pod_id> command. The
pod will be created with the initial state 'created'.

OPTIONS:
   --cgroup-parent value      Set parent cgroup for the pod
   --infra                    Create an infra container associated with the
pod to share namespaces with
   --infra-command value      The command to run on the infra container when
the pod is started (default: "/pause")
   --infra-image value        The image of the infra container to associate
with the pod (default: "k8s.gcr.io/pause:3.1")
   --label value, -l value    Set metadata on pod (default [])
   --label-file value         Read in a line delimited file of labels
(default [])
   --name value, -n value     Assign a name to the pod
   --pod-id-file value        Write the pod ID to the file
   --publish value, -p value  Publish a container's port, or a range of
ports, to the host (default [])
   --share value              A comma delimited list of kernel namespaces
the pod will share (default: "cgroup,ipc,net,uts")
```

가장 기본적인 `podman pod create` 컨텍스트에서 간단히 발행할 수 있으며 Podman은 추가 속성 없이 포드를 생성합니다. 또한 포드에 임의의 이름이 할당됩니다.

```
$ sudo podman pod create

9e0a57248aedc453e7b466d73ef769c99e35d265d97f6fa287442083246f3762
```

다음 `podman pod list`명령을 사용하여 포드를 나열할 수 있습니다 .

```
$ sudo podman pod list
POD ID         NAME             STATUS    CREATED     # OF CONTAINERS
INFRA ID
9e0a57248aed   youthful_jones  Running 5 seconds ago   1
6074ffd22b93
```

컨테이너에는 단일 컨테이너가 있습니다. 컨테이너는 `infra` 명령입니다. `podman ps`명령줄 스위치 *--pod*를 전달하여 명령을 사용하여 이를 추가로 관찰할 수 있습니다 .

```
$ sudo podman ps -a --pod
CONTAINER ID  IMAGE                    COMMAND  CREATED      STATUS     PORTS
NAMES               POD
6074ffd22b93  k8s.gcr.io/pause:3.1    3 minutes ago  Up 3 minutes ago
9e0a57248aed-infra  9e0a57248aed
```

여기에서 `podman ps` 의 POD ID가 `podman pod list`의 POD ID와 일치함 을 알 수 있습니다. 그리고 컨테이너 이미지는 기본 `infra` 컨테이너 이미지와 동일합니다.

Add a container to a pod

`podman create`및 `podman run`명령 에서 *`--pod`* 옵션을 사용하여 포드에 컨테이너를 추가할 수 있습니다. 예를 들어, 여기에서 **top**을 실행하는 컨테이너를 새로 생성된 *`youthful_jones`* 포드에 추가합니다. *`--pod`*의 사용에 주목하십시오.

```
$ sudo podman run -dt --pod youthful_jones docker.io/library/alpine:latest
top
0f62e6dcdfdbf3921a7d73353582fa56a545502c89f0dfcb8736ce7be61c9271
```

이제 포드에는 두 개의 컨테이너가 있습니다.

```
$ sudo podman pod ps
POD ID         NAME            STATUS  CREATED         # OF CONTAINERS
INFRA ID
9e0a57248aed   youthful_jones  Running 7 minutes ago   2
6074ffd22b93
```

컨테이너 목록을 보면 각 컨테이너와 해당 포드 할당도 볼 수 있습니다.

```
$ sudo podman ps -a --pod
CONTAINER ID  IMAGE                             COMMAND  CREATED
STATUS             PORTS  NAMES               POD
0f62e6dcdfdb  docker.io/library/alpine:latest  top      14 seconds ago  Up
14 seconds ago         awesome_archimedes  9e0a57248aed
6074ffd22b93  k8s.gcr.io/pause:3.1                      7 minutes ago   Up 7
minutes ago          9e0a57248aed-infra  9e0a57248aed
```

Shortcut to create pods

최근에`podman run` 및 `podman create`명령을 통해 포드를 생성하는 기능을 추가했습니다 . 이 접근

방식으로 포드를 생성할 때의 한 가지 장점은 컨테이너에 대해 선언된 일반 포트 바인딩이 infra 컨테이너에 자동으로 할당된다는 것입니다. 그러나 커널 네임스페이스 또는 다른 "infra컨테이너 이미지 사용과 같은 포드 생성을 위해 더 세분화된 옵션을 지정해야 하는 경우 처음에 설명한 대로 포드를 수동으로 생성해야 합니다. 그럼에도 불구하고 비교적 기본적인 포드 생성의 경우 바로 가기가 편리합니다. 이 기능은 최근에 추가되었기 때문에 Red Hat Enterprise Linux 7.6 및 8 Beta에 포함된 Podman 버전에서는 사용할 수 없습니다.

새 컨테이너로 새 포드를 만들려면 *--pod*: 를 전달하기만 하면 new:<name>됩니다. **new:**를 사용하면 기존 포드에 컨테이너를 할당하는 대신 새 포드를 생성하려는 Podman을 나타냅니다.

포드 내에 nginx 컨테이너를 생성하고 컨테이너의 포트 80을 호스트의 포트 32597에 노출하려면 다음을 수행합니다.

```
$ sudo podman run -dt --pod new:nginx -p 32597:80
quay.io/libpod/alpine_nginx:latest
ac8839fc7dead8e391e7983ad8d0c27ce311d190b0a8eb72dcde535de272d537

$ curl http://localhost:32597
podman rulez
```

다음은 컨테이너를 나열할 때의 모습입니다.

```
$ sudo podman ps -ap
CONTAINER ID  IMAGE                                 COMMAND
CREATED       STATUS            PORTS                   NAMES
POD
ac8839fc7dea  quay.io/libpod/alpine_nginx:latest  nginx -g daemon o...  4
minutes ago Up 4 minutes ago                        happy_cray
3e4cad88f8c2
c2f7c5651275  k8s.gcr.io/pause:3.1                                      4
minutes ago Up 4 minutes ago  0.0.0.0:32597->80/tcp  3e4cad88f8c2-infra
3e4cad88f8c2
```

Pods and container management

Podman에서 Pod와 해당 컨테이너의 상태는 서로 배타적일 수 있습니다. 즉, Pod의 상태에 영향을 주지 않고 Pod 내의 컨테이너를 다시 시작, 중지 및 시작할 수 있습니다. demodb라는 포드가 있고 MariaDB와 nginx 세션을 실행하는 두 개의 컨테이너(및 infra 컨테이너)가 포함되어 있다고 가정 합니다.

```
$ sudo podman pod ps
POD ID         NAME   STATUS    CREATED             # OF CONTAINERS   INFRA
ID
fa7924a5196c   demodb Running   About a minute ago  3
3005ed8491d0

$ sudo podman ps -p
CONTAINER ID  IMAGE                                 COMMAND
```

```
CREATED       STATUS             PORTS  NAMES               POD
02e37a3b9873  quay.io/libpod/alpine_nginx:latest   nginx -g daemon o... 4
minutes ago Up 4 minutes ago         optimistic_edison  fa7924a5196c
2597454063f8  quay.io/baude/mariadbpoddemo:latest  docker-entrypoint... 4
minutes ago Up 4 minutes ago         eloquent_golick    fa7924a5196c
```

nginx 컨테이너를 중지하고 시작하려는 경우 MariaDB 컨테이너 및 포드 자체의 상태는 변경되지 않은 상태로 유지됩니다.

```
$ sudo podman stop optimistic_edison
02e37a3b987300e9124b61820119ae425c5e496b907800ecaf1194a3f50e5dcc
```

nginx 컨테이너가 중지된 상태에서도 여전히 demopod실행 중이고 MariaDB 컨테이너가 변경되지 않은 상태로 유지되는 것을 볼 수 있습니다 .

```
$ sudo podman pod ps
POD ID         NAME   STATUS    CREATED       # OF CONTAINERS   INFRA ID
fa7924a5196c   demodb Running   5 minutes ago 3                 3005ed8491d0

$ sudo podman ps -p
CONTAINER ID  IMAGE                                COMMAND
CREATED       STATUS             PORTS  NAMES            POD
2597454063f8  quay.io/baude/mariadbpoddemo:latest  docker-entrypoint... 5
minutes ago Up 5 minutes ago         eloquent_golick  fa7924a5196c
```

그리고 nginx 컨테이너를 시작하여 포드를 원래 상태로 복원할 수 있습니다.

```
$ sudo podman start optimistic_edison
optimistic_edison

$ sudo podman ps -p
CONTAINER ID  IMAGE                                COMMAND
CREATED       STATUS             PORTS  NAMES               POD
02e37a3b9873  quay.io/libpod/alpine_nginx:latest   nginx -g daemon o... 8
minutes ago Up 6 seconds ago         optimistic_edison  fa7924a5196c
2597454063f8  quay.io/baude/mariadbpoddemo:latest  docker-entrypoint... 8
minutes ago Up 8 minutes ago         eloquent_golick    fa7924a5196c
```

podman pod stop명령을 사용하여 포드와 모든 컨테이너를 중지할 수도 있습니다 .

```
$ sudo podman pod stop demodb
fa7924a5196cb403298ad2ce24f0db30a3790e80729c7704ef5fdc27302f7ad0

$ sudo podman ps -ap
```

```
CONTAINER ID  IMAGE                                       COMMAND
CREATED        STATUS                       PORTS                     NAMES
POD
02e37a3b9873  quay.io/libpod/alpine_nginx:latest   nginx -g daemon o... 10
minutes ago Exited (0) 21 seconds ago
optimistic_edison   fa7924a5196c
2597454063f8  quay.io/baude/mariadbpoddemo:latest  docker-entrypoint... 10
minutes ago Exited (0) 19 seconds ago
eloquent_golick     fa7924a5196c
3005ed8491d0  k8s.gcr.io/pause:3.1                                     10
minutes ago Exited (0) 19 seconds ago  0.0.0.0:43871->3306/tcp
fa7924a5196c-infra  fa7924a5196c
```

그리고 Pod의 상태를 보면 Exited 상태로 표시됩니다.

```
$ sudo podman pod ps
POD ID         NAME     STATUS   CREATED        # OF CONTAINERS   INFRA ID
fa7924a5196c   demodb Exited   13 minutes ago   3
3005ed8491d0
```

마찬가지로 포드와 모든 컨테이너 백업을 시작할 수도 있습니다. 그런 다음 포드의 모든 컨테이너가 실행 중이어야 하며 포드는 Running 상태를 표시해야 합니다.

```
$ sudo podman pod start demodb
fa7924a5196cb403298ad2ce24f0db30a3790e80729c7704ef5fdc27302f7ad0

$ sudo podman ps -p
CONTAINER ID  IMAGE                                       COMMAND
CREATED        STATUS            PORTS  NAMES              POD
02e37a3b9873  quay.io/libpod/alpine_nginx:latest   nginx -g daemon o... 14
minutes ago Up 5 seconds ago         optimistic_edison  fa7924a5196c
2597454063f8  quay.io/baude/mariadbpoddemo:latest  docker-entrypoint... 14
minutes ago Up 4 seconds ago         eloquent_golick    fa7924a5196c

$ sudo podman pod ps
POD ID         NAME   STATUS    CREATED        # OF CONTAINERS  INFRA ID
fa7924a5196c   demodb Running   14 minutes ago 3                3005ed8491d0
```

`podman pod restart` Pod 내의 모든 컨테이너를 다시 시작 하는 명령 도 있습니다 .

참조링크

- https://developers.redhat.com/blog/2019/01/15/podman-managing-containers-pods

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

Permanent link:
https://atl.kr/dokuwiki/doku.php/podman_%EB%A1%9C%EC%BB%AC_%EC%BB%A8%ED%85%8C%EC%9D%B4%EB%84%88_%EB%9F%B0%ED%83%80%EC%9E%84%EC%97%90%EC%84%9C_%ED%8F%AC%EB%93%9C_%EB%B0%8F_%EC%BB%A8%ED%85%8C%EC%9D%B4%EB%84%88_%EA%B4%80%EB%A6%AC

Last update: **2021/07/27 01:57**