Linux Ubuntu 22.4.5 Manual Pages on command 'runc-spec.8'

$ man runc-spec.8

runc-spec (8"") runc-spec (8"")

NAME

runc-spec - create a new specification file

SYNOPSIS

runc spec [option ...]

DESCRIPTION

The spec command creates the new specification file named config.json for the bundle.

The spec generated is just a starter file. Editing of the spec is required to achieve desired results. For example, the newly generated spec includes an args parameter that is initially set to call the sh command when the container is started. Calling sh may work for an ubuntu container or busybox, but will not work for containers that do not include the sh binary.

OPTIONS

--bundle|-b path : Set path to the root of the bundle directory.

--rootless : Generate a configuration for a rootless container. Note this option is entirely different from the global --rootless option.

EXAMPLES

To run a simple "hello-world" container, one needs to set the args parameter in the spec to call hello. This can be done using sed(1), jq(1), or a text editor.

The following commands will:

- create a bundle for hello-world;

- change the command to run in a container to /hello using jq(1);

- run the hello command in a new hello-world container named container1.

```
mkdir hello
cd hello
docker pull hello-world
docker export $(docker create hello-world) > hello-world.tar
mkdir rootfs
tar -C rootfs -xf hello-world.tar
runc spec
jq '.process.args |= ["/hello"]' < config.json > new.json
mv -f new.json config.json
runc run container1
```

In the run command above, container1 is the name for the instance of the container that you are starting. The name you provide for the container instance must be unique on your host.

An alternative for generating a customized spec config is to use oci-runtime-tool; its sub-command oci-runtime-tool generate has lots of options that can be used to do any customizations as you want. See runtime-tools ?https://github.com/opencontainers/runtime-tools? to get more information.

When starting a container through runc, the latter usually needs root privileges. If not already running as root, you can use sudo(8), for example:

```
sudo runc start container1
```

Alternatively, you can start a rootless container, which has the ability to run without root privileges. For this to work, the specification file needs to be adjusted accordingly. You can pass the --rootless option to this command to generate a proper rootless spec file.

SEE ALSO

runc-run(8), runc(8).