

`slim` will build a VM from a Dockerfile. Slim works by building and extracting a rootfs from a Dockerfile, and packaging a corresponding kernel and initrd into a desired image.

This results in a real VM that can boot instantly, while using very limited resources—all with a couple of lines in a Dockerfile.

## Using slim

The following are a few ways you can use slim to build VM images.

### Build a Ubuntu Focal Cloud Raw VM Image

1. Provide a Dockerfile `images/ubuntu-20.04-cloud-init`

```
1 FROM ubuntu:20.04 AS kernel
2 RUN apt-get update && \
3     apt-get install -y linux-virtual && \
4     apt-get clean
5
6 FROM ubuntu:20.04
7
8 # Extract the kernel, modules, and initrd
9 COPY --from=kernel /lib/modules /lib/modules
10 COPY --from=kernel /boot/vmlinuz-* /vmlinuz
11 COPY --from=kernel /boot/initrd.img-* /initrd
12
13 RUN apt-get update
14 # Needed for configuring server and setting up devices.
15 RUN apt install cloud-init udev kmod -y
16 # If you'd like to be able to ssh in:
17 RUN apt install openssh-server sudo -y
```

2. Extract an initrd, rootfs, and uncompressed kernel.

```
1 $ slim build images/ubuntu-20.04-cloud-init
2 ...
3 $ ls -lh ~/.slim/registry/ubuntu-20.04-cloud-init
4 -rw-r--r-- 1 cjparnin staff 16M Jan 2 17:37 initrd
5 -rw-r--r-- 1 cjparnin staff 512M Jan 2 19:24 rootfs
6 -rw-----@ 1 cjparnin staff 29M Nov 5 12:04 vmlinuz
```

3. Provide a user-data and meta-data file to customize VM.

```

1 $ slim cloudinit images/ubuntu-20.04-cloud-init
2 ...
3 $ ls -lh ~/.slim/registry/ubuntu-20.04-cloud-init
4 -rw-r--r-- 1 cjparnin staff 366K Jan  2 21:19 cidata.iso

```

```

-----END SSH HOST KEY KEYS-----
[ 5.266420] cloud-init[403]: The system started in 5.26 seconds
[ OK ] Finished Execute cloud user/final scripts.
[ OK ] Reached target Cloud-init target.

Ubuntu 20.04.3 LTS basicvm hvc0

basicvm login:
basicvm login: ubuntu
Password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-91-generic aarch64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

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@basicvm:~$ ip -color a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
   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
2: enp0s1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
   link/ether d2:7e:11:b0:bf:c2 brd ff:ff:ff:ff:ff:ff
   inet 192.168.64.13/24 brd 192.168.64.255 scope global dynamic enp0s1
       valid_lft 86322sec preferred_lft 86322sec
   inet6 fde2:e351:3e11:7f27:d07e:11ff:feb0:bfc2/64 scope global dynamic mngtmpaddr nop
       valid_lft 2591986sec preferred_lft 604786sec
   inet6 fe80::d07e:11ff:feb0:bfc2/64 scope link
       valid_lft forever preferred_lft forever
ubuntu@basicvm:~$

```

VM Running in MAC M1 (arm64).

## Create a custom Alpine RAM only VM Image

1. Provide a Dockerfile and custom init script.

See images/alpine3.12-raw.

- 
2. Build initrd and kernel.

```
1 $ slim build images/alpine3.12-raw -f initrd
2 ...
3 $ ls -lh ~/.slim/registry/alpine3.12-raw
4 -rw-r--r-- 1 cjparnin staff 22M Jan 2 20:50 initrd
5 -rw-r--r-- 1 cjparnin staff 4.6M Dec 28 11:22 vmlinuz
```

### Create a Hyper-V VHD Image (Windows)

The following creates a Ubuntu Focal with cloud-init, but the necessary hyper-v kernel modules, and bootable image for Microsoft's Hyper-V.

1. Provide a Dockerfile.

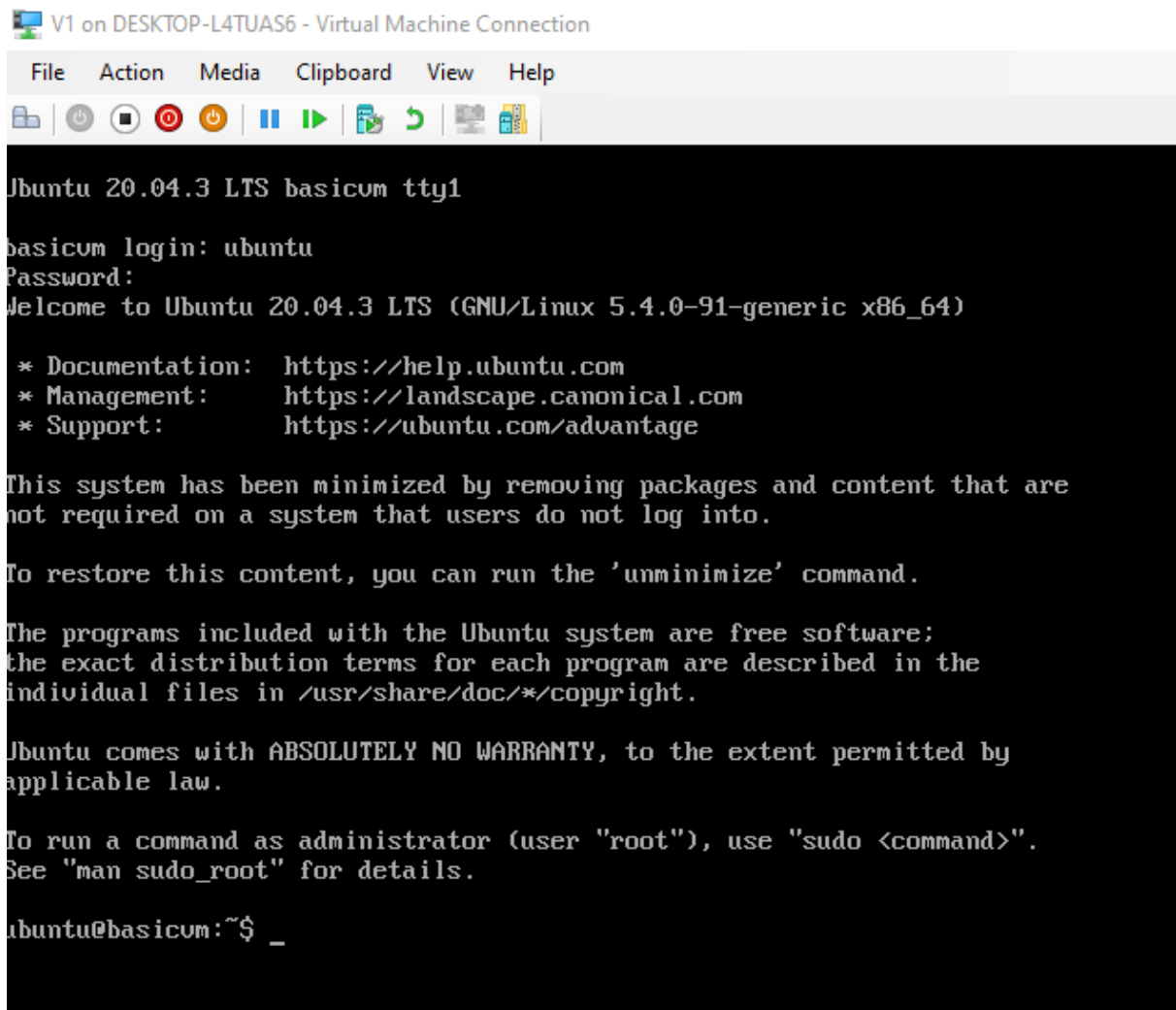
See images/ubuntu-20.04-ci-hyperv.

2. Create a VHD disk drive (1G) with EFI bootable partition.

```
1 PS slim build images/ubuntu-20.04-ci-hyperv -f vhd -s 1024
2 ...
3 PS ls ~/.slim/registry/ubuntu-20.04-ci-hyperv
4 -a---- 1/2/2022 3:19 PM 738381824 rootfs.vhd
```

3. Provide a user-data and meta-data file to customize VM.

```
1 $ slim cloudinit images/ubuntu-20.04-ci-hyperv
2 ...
3 $ ls -lh ~/.slim/registry/ubuntu-20.04-ci-hyperv
4 -a---- 1/1/2022 3:32 PM 374784 cidata.iso
```



## Installing slim

Simply clone this repo, cd slim, and run:

```
1 npm install
2 npm link
3
4 # Pull docker images used for system dependencies.
5 slim init
```

You must have docker on your system.