David Thomsen
SYS-265

```
PS C:\Users\david-thomsen-adm> ssh david@docker01-david.david.local
The authenticity of host 'docker01-david.david.local (10.0.5.12)' can't be established.
ECDSA key fingerprint is SHA256:INEH1XDCgeuzRz8QNkatkDx6S3TQuDDCCeQvnRiwoH8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'docker01-david.david.local,10.0.5.12' (ECDSA) to the list of known hosts.
david@docker01-david.david.local's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-94-generic x86_64)Welcome to Ubuntu 20.04.3 LTS (GNU/Linu
x 5.4.0-94-generic x86_64)

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

  System information as of Sun 13 Feb 2022 07:01:38 PM UTC

  System load:  0.48              Processes:               215
  Usage of /:   45.8% of 9.78GB   Users logged in:         1
  Memory usage: 27%               IPv4 address for ens160: 10.0.5.12
  Swap usage:   0%


45 updates can be applied immediately.
34 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable
```

```
$ sudo -i
sudo: unable to resolve host docker01-david: Temporary failure in name resolution
[sudo] password for david:
root@docker01-david:~# ping -c 1 champlain.edu
PING champlain.edu (208.115.107.132) 56(84) bytes of data.
64 bytes from 208-115-107-132-reverse.wowrack.com (208.115.107.132): icmp_seq=1 ttl=48 time=78.2 ms

--- champlain.edu ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 78.224/78.224/78.224/0.000 ms
root@docker01-david:~#
```

```
root@docker01-david:~# systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2022-02-13 23:05:33 UTC; 15s ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 19692 (dockerd)
      Tasks: 7
     Memory: 29.4M
     CGroup: /system.slice/docker.service
             └─19692 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
```

```
$ docker version
Client: Docker Engine - Community
 Version:           20.10.12
 API version:       1.41
 Go version:        go1.16.12
 Git commit:        e91ed57
 Built:             Mon Dec 13 11:45:33 2021
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:          20.10.12
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.16.12
  Git commit:       459d0df
  Built:            Mon Dec 13 11:43:42 2021
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.4.12
  GitCommit:        7b11cfaabd73bb80907dd23182b9347b4245eb5d
 runc:
  Version:          1.0.2
  GitCommit:        v1.0.2-0-g52b36a2
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
```

(The local admin user doesnt have a default directory so it just says $, is part of the docker group though)

```
 GitCommit:        de40ad0
$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:97a379f4f88575512824f3b352bc03cd75e239179eea0fecc38e597b2209f49a
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

```
root@docker01-david:/# chmod +x /usr/local/bin/docker-compose
root@docker01-david:/# docker-compose --version
docker-compose version 1.27.4, build 40524192
root@docker01-david:/#
```

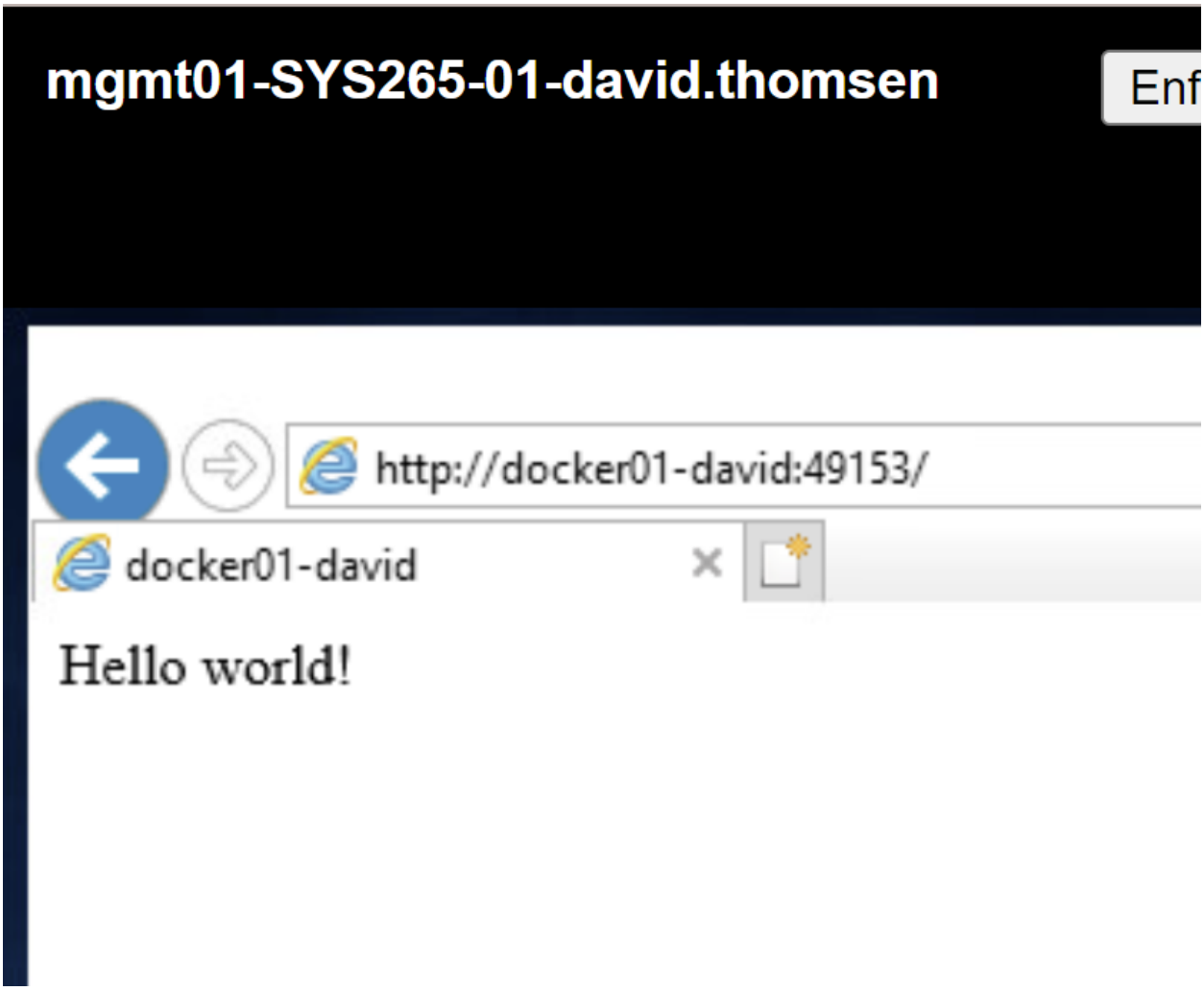
```
root@docker01-david:/# docker run --rm archlinux:latest /bin/echo "HELLO SYS265 SNOWY DAYS"
Unable to find image 'archlinux:latest' locally
latest: Pulling from library/archlinux
ac75e2c22f5d: Pull complete
e9c7cf39659c: Pull complete
Digest: sha256:14b7f97a324bf9988a04074905d2c3333847012a4ba610d8d5e4c27400ee6377
Status: Downloaded newer image for archlinux:latest
HELLO SYS265 SNOWY DAYS
root@docker01-david:/#
```

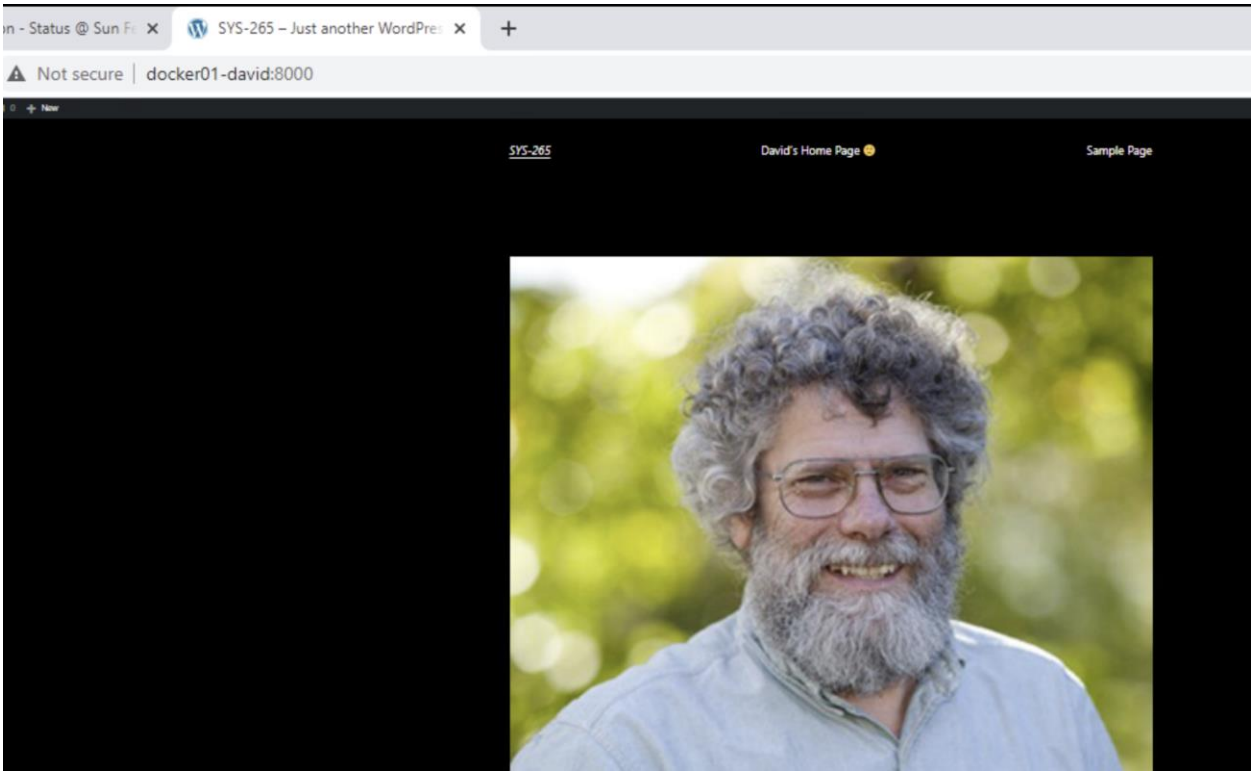
```
root@docker01-david:/# cat /etc/lsb-release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=20.04
DISTRIB_CODENAME=focal
DISTRIB_DESCRIPTION="Ubuntu 20.04.3 LTS"
root@docker01-david:/# echo "Current Kernal is: $(uname -a)"
Current Kernal is: Linux docker01-david 5.4.0-94-generic #106-Ubuntu SMP Thu Jan 6 23:58:14 UTC 2022
 x86_64 x86_64 x86_64 GNU/Linux
root@docker01-david:/# docker run -it archlinux /bin/uname -a
Linux d5572e978026 5.4.0-94-generic #106-Ubuntu SMP Thu Jan 6 23:58:14 UTC 2022 x86_64 GNU/Linux
root@docker01-david:/#
```

The Kernal was pulled from outside of the Container

```
   -d, --detach=true|false
      Detached mode: run the container in the background and print the new container ID.  The
   default is false.

   At  any  time  you can run docker ps in the other shell to view a list of the running con-
   tainers. You can reattach to a detached container with docker attach.

   When attached in the tty mode, you can detach from the container (and  leave  it  running)
   using  a  configurable key sequence. The default sequence is CTRL-p CTRL-q.  You configure
   the key sequence using the --detach-keys option or  a  configuration  file.   See  config-
   json(5) for documentation on using a configuration file.
```

```
-P, --publish-all=true|false
   Publish all exposed ports to random ports on the host interfaces. The default is false.

When  set  to true publish all exposed ports to the host interfaces. The default is false.
If the operator uses -P (or -p) then Docker will make the exposed port accessible  on  the
host and the ports will be available to any client that can reach the host. When using -P,
Docker will bind any exposed port to a random port on the host within  an  ephemeral  port
range  defined  by /proc/sys/net/ipv4/ip_local_port_range. To find the mapping between the
host ports and the exposed ports, use docker port(1).
```


mgmt01-SYS265-01-david.thomsen
Enf
http://docker01-david:49153/
docker01-david
Hello world!


- Status @ Sun
SYS-265 – Just another WordPre
Not secure
docker01-david:8000
New
SYS-265
David's Home Page
Sample Page