



Lab 4 - DEVOPS & BLKCHAIN x Install Terraform | Terraform | H x BCDV-4033/terraform/demo.m x +

https://developer.hashicorp.com/terraform/tutorials/docker-get-started/install-dli

faizan@DESKTOP-S27V5UT: ~

```
faizan@DESKTOP-S27V5UT:~$ echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] \
> https://apt.releases.hashicorp.com $(lsb_release -cs) main" | \
> sudo tee /etc/apt/sources.list.d/hashicorp.list
deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com jammy main
faizan@DESKTOP-S27V5UT:~$ sudo apt update
Hit:1 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:6 https://apt.releases.hashicorp.com jammy InRelease [12.9 kB]
Get:7 https://apt.releases.hashicorp.com jammy/main amd64 Packages [121 kB]
Fetched 134 kB in 1s (143 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
15 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: https://download.docker.com/linux/ubuntu/dists/jammy/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt
/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
faizan@DESKTOP-S27V5UT:~$ sudo apt-get install terraform
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  terraform
0 upgraded, 1 newly installed, 0 to remove and 15 not upgraded.
Need to get 26.9 MB of archives.
After this operation, 84.5 MB of additional disk space will be used.
Get:1 https://apt.releases.hashicorp.com jammy/main amd64 terraform amd64 1.7.4-1 [26.9 MB]
Fetched 26.9 MB in 3s (7706 kB/s)
Selecting previously unselected package terraform.
(Reading database ... 43049 files and directories currently installed.)
Preparing to unpack .../terraform_1.7.4-1_amd64.deb ...
Unpacking terraform (1.7.4-1) ...
```

We use cookies & other similar technology

AutoSave 00 Document1 - Word Search Faizan UI Haq FU

File Home Insert Draw Design Layout References Mailings Review View Help

Clipboard Font

faizan@DESKTOP-S27V5UT: ~

```
-generate-config-out-path (Experimental) If import blocks are present in
configuration, instructs Terraform to generate HCL
for any imported resources not already present. The
configuration is written to a new file at PATH,
which must not already exist. Terraform may still
attempt to write configuration if the plan errors.

-input=true Ask for input for variables if not directly set.

-lock=false Don't hold a state lock during the operation. This
is dangerous if others might concurrently run
commands against the same workspace.

-lock-timeout=0s Duration to retry a state lock.

-no-color If specified, output won't contain any color.

-out-path Write a plan file to the given path. This can be
used as input to the "apply" command.

-parallelism=n Limit the number of concurrent operations. Defaults
to 10.

-state=statefile A legacy option used for the local backend only.
See the local backend's documentation for more
information.
faizan@DESKTOP-S27V5UT:~$ terraform -v
Command 'terraform' not found, did you mean:
  Command 'terraform' from snap terraform (1.7.4)
See 'snap info <snapname>' for additional versions.
faizan@DESKTOP-S27V5UT:~$ terraform -v
Terraform v1.7.4
on linux_amd64
faizan@DESKTOP-S27V5UT:~$
```

Page 2 of 2 0 words English (Pakistan) Text Predictions: On Accessibility: Investigate

Type here to search 4°C 9:16 pm 07/03/2024

Lab 4 - DEVOPS & BL | x | Install Terraform | Ter | x | BCDV-4033/terraform | x | (70) pitbull - YouTube | x | (70) Pitbull - Give Me | x | Really Great Tax Filing | x | +

https://developer.hashicorp.com/terraform/tutorials/docker-get-started/install-dli

faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container

faizan@DESKTOP-S27V5UT:~/WSL\$ mkdir learn-terraform-docker-container
faizan@DESKTOP-S27V5UT:~/WSL\$ cd learn-terraform-docker-container/
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container\$ touch main.tf
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container\$ code .
Installing VS Code Server For x64 (903b1e9d8990623e3d7da1df3d33db3e42d80eda)
Downloading: 12%
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container\$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "~> 3.0.1"...
\$ - Installing kreuzwerker/docker v3.0.2...
\$ - Installed kreuzwerker/docker v3.0.2 (self-signed, key ID BD080C4571C6104C)
Partner and community providers are signed by their developers.
If you'd like to know more about provider signing, you can read about it here:
https://www.terraform.io/docs/cli/plugins/signing.html
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.
Terraform has been successfully initialized!
You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.
If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container\$

We use cookies & other similar technology to collect data to improve your experience on our site, as described in our [Privacy Policy](#) and [Cookie Policy](#). Manage Preferences ACCEPT

Type here to search

File Edit Selection View Go Run ... Search

Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

main.tf x

Ubuntu > home > faizan > WSL > learn-terraform-docker-container

```
1 terraform {  
2   required_providers {  
3     docker = {  
4       source = "kreuzwerker/docker"  
5       version = "~> 3.0.1"  
6     }  
7   }  
8 }  
9  
10 provider "docker" {}  
11  
12 resource "docker_image" "nginx" {  
13   name = "nginx"  
14   keep_locally = false  
15 }  
16  
17 resource "docker_container" "nginx" {  
18   image = docker_image.nginx.image_id  
19   name = "tutorial"  
20  
21   ports {  
22     internal = 80  
23     external = 8000  
24   }  
25 }  
26
```

faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container\$ terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

```
# docker_container.nginx will be created  
+ resource "docker_container" "nginx" {  
+ attach = false  
+ bridge = (known after apply)  
+ command = (known after apply)  
+ container_logs = (known after apply)  
+ container_read_refresh_timeout_milliseconds = 15000  
+ entrypoint = (known after apply)  
+ env = (known after apply)  
+ exit_code = (known after apply)  
+ hostname = (known after apply)  
+ id = (known after apply)  
+ image = (known after apply)  
+ init = (known after apply)  
+ ipc_mode = (known after apply)  
+ log_driver = (known after apply)  
+ logs = false  
+ must_run = true  
+ name = "tutorial"  
+ network_data = (known after apply)  
+ read_only = false  
+ remove_volumes = true  
+ restart = "no"  
+ rm = false  
+ runtime = (known after apply)  
+ security_opts = (known after apply)
```

Ln 26, Col 1 Spaces: 4 UTF-8 CRLF Plain Text

Type here to search

3°C 9:39 pm 07/03/2024

```
File Edit Selection View Go Run ... Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

main.tf x
Ubuntu > home > faizan > WSL > learn-terraform-docker-container
1 terraform {
2   required_providers {
3     docker = {
4       source = "kreuzwerker/docker"
5       version = "~> 3.0.1"
6     }
7   }
8 }
9
10 provider "docker" {}
11
12 resource "docker_image" "nginx" {
13   name = "nginx"
14   keep_locally = false
15 }
16
17 resource "docker_container" "nginx" {
18   image = docker_image.nginx.image_id
19   name = "tutorial"
20
21   ports {
22     internal = 80
23     external = 8080
24   }
25 }
26
```

```
faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container
+ internal = 80
+ ip = "0.0.0.0"
+ protocol = "tcp"
}

# docker_image.nginx will be created
+ resource "docker_image" "nginx" {
+   id = (known after apply)
+   image_id = (known after apply)
+   keep_locally = false
+   name = "nginx"
+   repo_digest = (known after apply)
}

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
Enter a value: yes

docker_image.nginx: Creating...
docker_image.nginx: Still creating... [10s elapsed]
docker_image.nginx: Creation complete after 18s [id=sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666anginx]
docker_container.nginx: Creating...
docker_container.nginx: Still creating... [10s elapsed]
docker_container.nginx: Creation complete after 11s [id=7b0943abe7a63afe7742561e90b6ee3ec33c7d8594516e04f5d26f41c8c317b1]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container$
```

```
File Edit Selection View Go Run ... Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

main.tf x
Ubuntu > home > faizan > WSL > learn-terraform-docker-container
1 terraform {
2   required_providers {
3     docker = {
4       source = "kreuzwerker/docker"
5       version = "~> 3.0.1"
6     }
7   }
8 }
9
10 provider "docker" {}
11
12 resource "docker_image" "ubuntu" {
13   name = "ubuntu" # Change t
14   keep_locally = false
15 }
16
17 resource "docker_container" "nginx" {
18   image = docker_image.ubuntu.image_i
19   name = "tutorial"
20
21   ports {
22     internal = 80
23     external = 8080
24   }
25 }
26
```

```
faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
7b0943abe7a6   e4720093a3c1   "/docker-entrypoint..." About a minute ago Up About a minute   0.0.0.0:8080->80/tcp
tutorial

faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container$ terraform plan
docker_image.nginx: Refreshing state... [id=sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666anginx]
docker_container.nginx: Refreshing state... [id=7b0943abe7a63afe7742561e90b6ee3ec33c7d8594516e04f5d26f41c8c317b1]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
+ create
- destroy
-/ destroy and then create replacement

Terraform will perform the following actions:

# docker_container.nginx must be replaced
-/+ resource "docker_container" "nginx" {
+   bridge = (known after apply)
+   command = [
-   "nginx",
-   "-g",
-   "daemon off;",
+   ] -> (known after apply)
+   container_logs = (known after apply)
+   cpu_shares = 0 -> null
+   dns = [] -> null
+   dns_opts = [] -> null
+   dns_search = [] -> null
+   entrypoint = [

```

```
faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container
} -> (known after apply)
- network_mode = "default" -> null
- privileged = false -> null
- publish_all_ports = false -> null
- runtime = "runc" -> (known after apply)
- security_opts = [] -> (known after apply)
- shm_size = 64 -> (known after apply)
- stop_signal = "SIGQUIT" -> (known after apply)
- stop_timeout = 0 -> (known after apply)
- storage_opts = {} -> null
- sysctls = {} -> null
- tmpfs = {} -> null
# (13 unchanged attributes hidden)
# (1 unchanged block hidden)
}

# docker_image.nginx will be destroyed
# (because docker_image.nginx is not in configuration)
- resource "docker_image" "nginx" {
  - id = "sha256:e4728093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a" -> null
  - image_id = "sha256:e4728093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a" -> null
  - keep_locally = false -> null
  - name = "nginx" -> null
  - repo_digest = "nginx@sha256:c26ae7472d624ba1fafd296e73cecc4f93f853088e6a9c13c0d52f6ca5865107" -> null
}

# docker_image.ubuntu will be created
+ resource "docker_image" "ubuntu" {
+ id = (known after apply)
+ image_id = (known after apply)
+ keep_locally = false
+ name = "ubuntu"
+ repo_digest = (known after apply)
}

Plan: 2 to add, 0 to change, 2 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if
you run "terraform apply" now.
faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container$

main.tf x
Ubuntu > home > faizan > WSL > learn-terraform-docker
1 terraform {
2   required_providers {
3     docker = {
4       source = "kreuzwerker/docker"
5       version = ">= 3.0.1"
6     }
7   }
8 }
9
10 provider "docker" {}
11
12 resource "docker_image" "nginx" {
13   name = "nginx:latest"
14   keep_locally = false
15 }
16
17 resource "docker_container" "nginx" {
18   image = docker_image.nginx.image_id
19   name = "tutorial"
20   ports {
21     internal = 80
22     external = 8090
23   }
24 }
25

faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container$ terraform plan
docker_image.nginx: Refreshing state... [id=sha256:e4728093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a]
docker_container.nginx: Refreshing state... [id=627ccde95c84e0d207f355069e2cef97fcfa2a281274344391af8d1957a38844]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
+ create

Terraform will perform the following actions:

# docker_container.nginx will be created
+ resource "docker_container" "nginx" {
  + attach = false
  + bridge = (known after apply)
  + command = (known after apply)
  + container_logs = (known after apply)
  + container_read_refresh_timeout_milliseconds = 15000
  + endpoint = (known after apply)
  + env = (known after apply)
  + exit_code = (known after apply)
  + hostname = (known after apply)
  + id = (known after apply)
  + image = "sha256:e4728093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a"
  + init = (known after apply)
  + ipc_mode = (known after apply)
  + log_driver = (known after apply)
  + logs = false
  + must_run = true
  + name = "tutorial"
  + network_data = (known after apply)
}
```

```
File Edit Selection View Go Run ... Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

main.tf x
Ubuntu > home > faizan > WSL > learn-terraform-docker-
1 terraform {
2   required_providers {
3     docker = {
4       source = "kreuzwerker/docker"
5       version = "~> 3.0.1"
6     }
7   }
8 }
9
10 provider "docker" {}
11
12 resource "docker_image" "nginx" {
13   name = "nginx:latest"
14   keep_locally = false
15 }
16
17 resource "docker_container" "nginx" {
18   image = docker_image.nginx.image_id
19   name = "tutorial"
20   ports {
21     internal = 80
22     external = 8090
23   }
24 }
25

faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container
+ network_data = (known after apply)
+ read_only = false
+ remove_volumes = true
+ restart = "no"
+ rm = false
+ runtime = (known after apply)
+ security_opts = (known after apply)
+ shm_size = (known after apply)
+ start = true
+ stdin_open = false
+ stop_signal = (known after apply)
+ stop_timeout = (known after apply)
+ tty = false
+ wait = false
+ wait_timeout = 60

+ ports {
+   external = 8090
+   internal = 80
+   ip = "0.0.0.0"
+   protocol = "tcp"
+ }

Plan: 1 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container$
```

```
File Edit Selection View Go Run ... Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

main.tf x
Ubuntu > home > faizan > WSL > learn-terraform-docker-
1 terraform {
2   required_providers {
3     docker = {
4       source = "kreuzwerker/docker"
5       version = "~> 3.0.1"
6     }
7   }
8 }
9
10 provider "docker" {}
11
12 resource "docker_image" "nginx" {
13   name = "nginx:latest"
14   keep_locally = false
15 }
16
17 resource "docker_container" "nginx" {
18   image = docker_image.nginx.image_id
19   name = "tutorial"
20   ports {
21     internal = 80
22     external = 8090
23   }
24 }
25

Select faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container
+ restart = "no"
+ rm = false
+ runtime = (known after apply)
+ security_opts = (known after apply)
+ shm_size = (known after apply)
+ start = true
+ stdin_open = false
+ stop_signal = (known after apply)
+ stop_timeout = (known after apply)
+ tty = false
+ wait = false
+ wait_timeout = 60

+ ports {
+   external = 8090
+   internal = 80
+   ip = "0.0.0.0"
+   protocol = "tcp"
+ }

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

docker_container.nginx: Creating...
docker_container.nginx: Creation complete after 4s [id=025613e90179e4e273bb2cca7ddd5fc3804ee8de88775e908ae9842f1042362b]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container$
```

```
faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container
/* resource "docker_image" "nginx" {
  ~ id      = "sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a" -> (known after apply)
  ~ image_id = "sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a" -> (known after apply)
  ~ name     = "nginx:latest" -> "nginx" # forces replacement
  ~ repo_digest = "nginx@sha256:c26ae7472d624ba1fafd296e73cecc4f93f853088e6a9c13c0d52f6ca5865107" -> (known after apply)
  # (1 unchanged attribute hidden)
}

# null_resource.remove_nginx will be created
+ resource "null_resource" "remove_nginx" {
+   id      = (known after apply)
+   triggers = {
+     "image_id" = (known after apply)
+   }
+ }

Plan: 3 to add, 0 to change, 2 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

docker_container.nginx: Destroying... [id=025613e90179e4e273bb2cca7dd5fc3804ee8de88775e908ae9842f1042362b]
docker_container.nginx: Destruction complete after 1s
docker_image.nginx: Destroying... [id=sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a]
docker_image.nginx: Destruction complete after 0s
docker_image.nginx: Creating...
docker_image.nginx: Still creating... [10s elapsed]
docker_image.nginx: Still creating... [20s elapsed]
docker_image.nginx: Creation complete after 22s [id=sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a]
null_resource.remove_nginx: Creating...
null_resource.remove_nginx: Provisioning with 'local-exec'...
null_resource.remove_nginx (local-exec): Executing: ["/bin/sh" "-c" "docker rmi -f sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a"]
docker_container.nginx: Creating...
null_resource.remove_nginx (local-exec): Untagged: nginx:latest
null_resource.remove_nginx (local-exec): Untagged: nginx@sha256:c26ae7472d624ba1fafd296e73cecc4f93f853088e6a9c13c0d52f6ca5865107
null_resource.remove_nginx (local-exec): Deleted: sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a
null_resource.remove_nginx: Creation complete after 3s [id=6862858865086089499]
docker_container.nginx: Creation complete after 7s [id=15f12bd62d07cd03bce343294984cfc8dfae74bffa7c8144f775f33ac8100f0]
Apply complete! Resources: 3 added, 0 changed, 2 destroyed.
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container$
```

```
faizan@DESKTOP-S27V5UT: ~/WSL/learn-terraform-docker-container
- security_opts = [] -> null
- shm_size      = 64 -> null
- start         = true -> null
- stdin_open    = false -> null
- stop_signal   = "SIGQUIT" -> null
- stop_timeout  = 0 -> null
- storage_opts  = {} -> null
- sysctls       = {} -> null
- tmpfs         = {} -> null
- tty           = false -> null
- wait          = false -> null
- wait_timeout  = 60 -> null

- ports {
-   external = 8000 -> null
-   internal = 80 -> null
-   ip       = "0.0.0.0" -> null
-   protocol = "tcp" -> null
- }

# null_resource.remove_nginx will be destroyed
- resource "null_resource" "remove_nginx" {
-   id      = "6862858865086089499" -> null
-   triggers = {
-     "image_id" = "sha256:e4720093a3c1381245b53a5a51b417963b3c4472d3f47fc301930a4f3b17666a"
-   } -> null
- }

Plan: 0 to add, 0 to change, 2 to destroy.

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

null_resource.remove_nginx: Destroying... [id=6862858865086089499]
null_resource.remove_nginx: Destruction complete after 0s
docker_container.nginx: Destroying... [id=15f12bd62d07cd03bce343294984cfc8dfae74bffa7c8144f775f33ac8100f0]
docker_container.nginx: Destruction complete after 1s

Destroy complete! Resources: 2 destroyed.
faizan@DESKTOP-S27V5UT:~/WSL/learn-terraform-docker-container$
```