



the essential Terraform Cheatsheet

by justin o'connor

general commands

get the terraform version
terraform version

download and update root modules
terraform get -update=true

open up a terraform interactive terminal
terraform console

create a dot diagram of terraform dependencies
terraform graph | dot -Tpng > graph.png

format terraform code to HCL standards
terraform fmt

validate terraform code syntax
terraform validate

enable tab auto-completion in the terminal
terraform -install-autocomplete

show information about provider requirements
terraform providers

login and logout of terraform cloud
terraform login and terraform logout

workspaces

list the available workspaces
terraform workspace list

create a new workspace
terraform workspace new development

select an existing workspace
terraform workspace select default

initilize terraform

initialize terraform in the current working directory
terraform init

skip plugin installation
terraform init -get-plugins=false

force plugin installation from a directory
terraform init -plugin-dir=PATH

upgrade modules and plugins at initialization
terraform init -upgrade

update backend configuration
terraform init -migrate-state -force-copy

skip backend configuration
terraform init -backend=false

use a local backend configuration
terraform init -backend-config=FILE

change state lock timeout (default is zero seconds)
terraform init -lock-timeout=120s

plan terraform

produce a plan with diff between code and state
terraform plan

output a plan file for reference during apply
terraform plan -out current.tfplan

output a plan to show effect of terraform destroy
terraform plan -destroy

target a specific resource for deployment
terraform plan -target=ADDRESS

note that the -target option is also available for the terraform apply and terraform destroy commands.

outputs

list available outputs
terraform output

output a specific value
terraform output NAME

apply terraform

apply the current state of terraform code
terraform apply

specify a previously generated plan to apply
terraform apply current.tfplan

enable auto-approval or automation
terraform apply -auto-approve

destroy terraform

destroy resources managed by terraform state
terraform destroy

enable auto-approval or automation
terraform destroy -auto-approve

manage terraform state

list all resources in terraform state
terraform state list

show details about a specific resource
terraform state show ADDRESS

track an existing resource in state under new name
terraform state mv SOURCE DESTINATION

import a manually created resource into state
terraform state import ADDRESS ID

pull state and save to a local file
terraform state pull > terraform.tfstate

push state to a remote location
terraform state push PATH

replace a resource provider
terraform state replace-provider A B

taint a resource to force redeployment on apply
terraform taint ADDRESS

untaint a previously tainted resource
terraform untaint ADDRESS