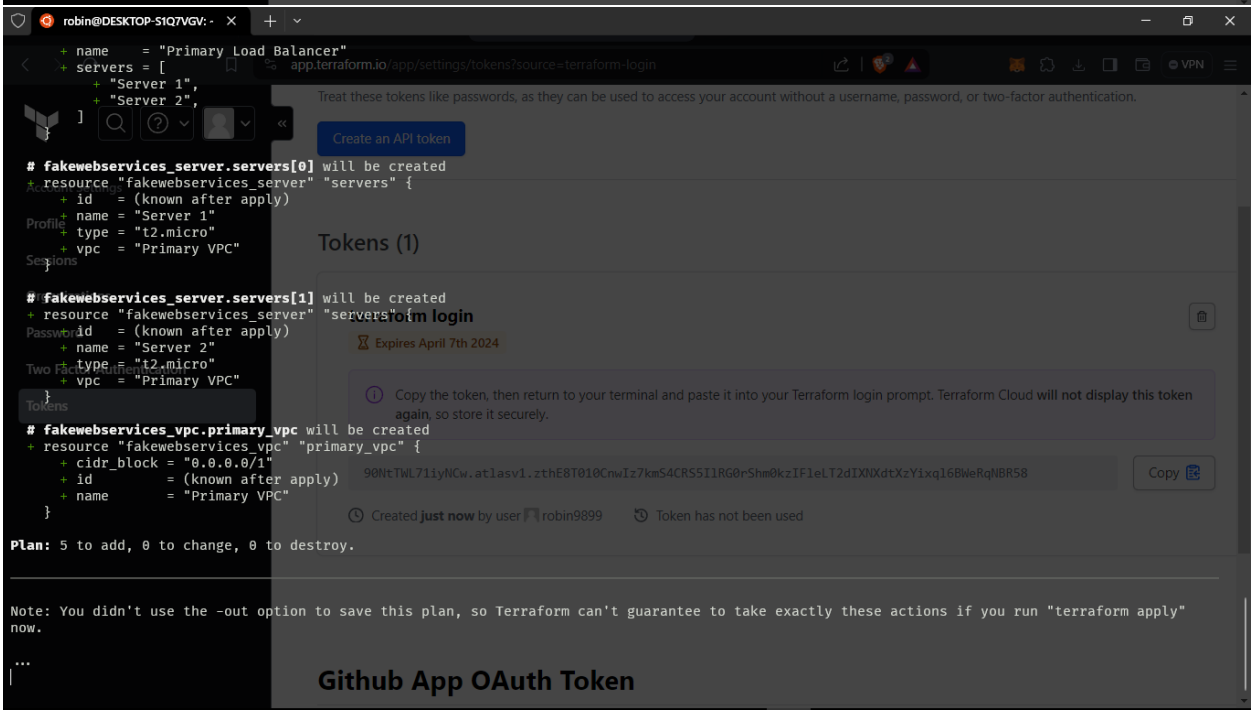
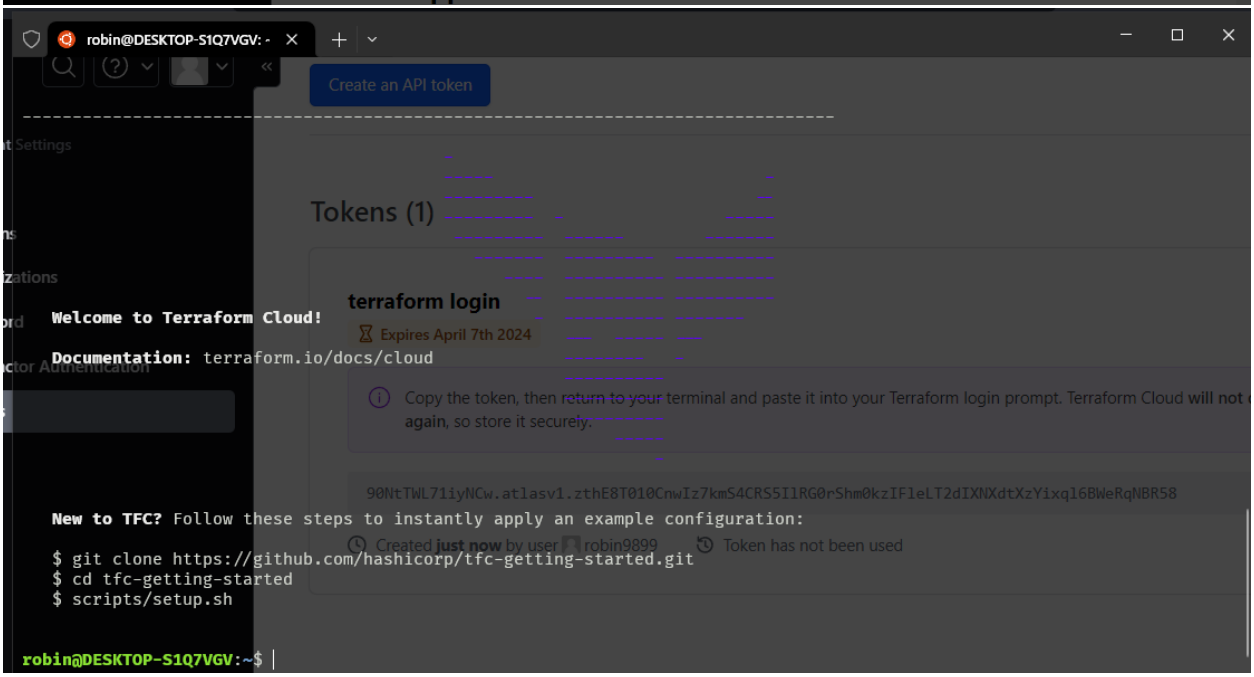
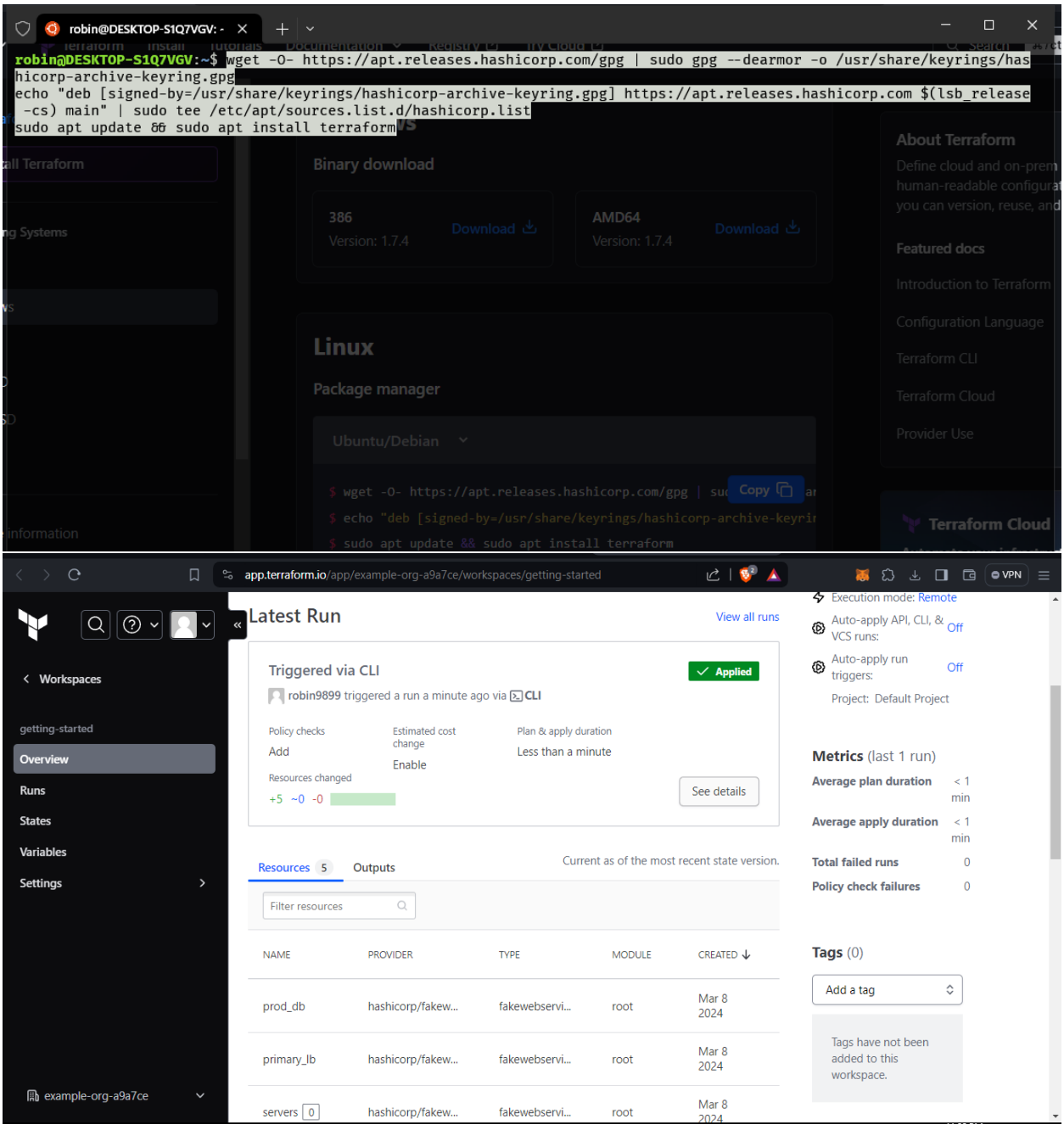

fakewebservices_database.prod_db: Creating...
fakewebservices_database.prod_db: Creation complete after 0s [id=fakedb-BHYAEwSGbvvsun2f]
fakewebservices_vpc.primary_vpc: Creation complete after 0s [id=fakevpc-fp1XVfuJN86xLwmt]
fakewebservices_server.servers[1]: Creating...
fakewebservices_server.servers[0]: Creating...
fakewebservices_server.servers[0]: Creation complete after 0s [id=fakeserver-XA7HreFMo7evAi27]
fakewebservices_server.servers[1]: Creation complete after 0s [id=fakeserver-qmN2ndUNdcsqo6bk]
fakewebservices_load_balancer.primary_lb: Creating...
fakewebservices_load_balancer.primary_lb: Creation complete after 0s [id=fakelb-wspRMEEBxUdeoKW4]
Apply complete! Resources: 5 added, 0 changed, 0 destroyed.
...
Tokens (1)
terraform login
Expires April 7th 2024
Copy the token, then return to your terminal and paste it into your Terraform login prompt. Terraform Cloud will not display this token again, so store it securely.
Copy
Token has not been used
==========
You did it! You just provisioned infrastructure with Terraform Cloud!
The organization we created here has a 30-day free trial of the Team &
Governance tier features. After the trial ends, you'll be moved to the Free tier.
You now have:
* Workspaces for organizing your infrastructure. Terraform Cloud manages
infrastructure collections with workspaces instead of directories. You
can view your workspace here:
https://app.terraform.io/app/example-org-a9a7ce/workspaces/getting-started
* Remote state management, with the ability to share outputs across
workspaces. We've set up state management for you in your current
workspace, and you can reference state from other workspaces using
the 'terraform_remote_state' data source.
* Much more!
To see the mock infrastructure you just provisioned and continue exploring
Terraform Cloud, visit:
https://app.terraform.io/fake-web-services
robin@DESKTOP-S1Q7VGV:~/tfc-getting-started$
Github App OAuth Token


robin@DESKTOP-S1Q7VGV: ~
+ name = "Primary Load Balancer"
+ servers = [
+ "Server 1",
+ "Server 2",
]
}
# fakewebservices_server.servers[0] will be created
+ resource "fakewebservices_server" "servers" {
+ id = (known after apply)
+ name = "Server 1"
+ type = "t2.micro"
+ vpc = "Primary VPC"
}
# fakewebservices_server.servers[1] will be created
+ resource "fakewebservices_server" "servers" {
+ id = (known after apply)
+ name = "Server 2"
+ type = "t2.micro"
+ vpc = "Primary VPC"
}
# fakewebservices_vpc.primary_vpc will be created
+ resource "fakewebservices_vpc" "primary_vpc" {
+ cidr_block = "0.0.0.0/1"
+ id = (known after apply)
+ name = "Primary VPC"
}
Plan: 5 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.
...
Tokens (1)
terraform login
Expires April 7th 2024
Copy the token, then return to your terminal and paste it into your Terraform login prompt. Terraform Cloud will not display this token again, so store it securely.
Copy
Token has not been used
Github App OAuth Token


robin@DESKTOP-S1Q7VGV:~$ cd tfc-getting-started
robin@DESKTOP-S1Q7VGV:~/tfc-getting-started$ scripts/setup.sh
Getting Started with Terraform Cloud
Terraform Cloud offers secure, easy-to-use remote state management and allows
you to run Terraform remotely in a controlled environment. Terraform Cloud runs
can be performed on demand or triggered automatically by various events.
This script will set up everything you need to get started. You'll be
applying some example infrastructure - for free - in less than a minute.
First, we'll do some setup and configure Terraform to use Terraform Cloud.
Press any key to continue (ctrl-c to quit):
Creating an organization and workspace ...
Writing Terraform Cloud configuration to backend.tf ...
Ready to go; the example configuration is set up to use Terraform Cloud!
An example workspace named 'getting-started' was created for you.
You can view this workspace in the Terraform Cloud UI here:
https://app.terraform.io/app/example-org-a9a7ce/workspaces/getting-started
Next, we'll run 'terraform init' to initialize the backend and providers:
$ terraform init
Press any key to continue (ctrl-c to quit):
terraform login
Expires April 7th 2024
Copy the token, then return to your terminal and paste it into your Terraform login prompt. Terraform Cloud will not display this token again, so store it securely.
Copy
Created just now by user robin9899
Token has not been used
Github App OAuth Token


robin@DESKTOP-S1Q7VGV: ~
Create an API token
Tokens (1)
terraform login
Expires April 7th 2024
Copy the token, then return to your terminal and paste it into your Terraform login prompt. Terraform Cloud will not
again, so store it securely.
90NtTWL71iyNCw.atlasv1.zthE8T010CnwIz7kmS4CRS5I1RG0rShm0kzIF1eLT2dIXNXdtXzYixql6BWeRqNBR58
Created just now by user robin9899
Token has not been used
Welcome to Terraform Cloud!
Documentation: terraform.io/docs/cloud
New to TFC? Follow these steps to instantly apply an example configuration:
$ git clone https://github.com/hashicorp/tfc-getting-started.git
$ cd tfc-getting-started
$ scripts/setup.sh
robin@DESKTOP-S1Q7VGV:~$


robin@DESKTOP-S1Q7VGV: ~
robin@DESKTOP-S1Q7VGV:~$ wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
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
sudo apt update && sudo apt install terraform
Binary download
386
Version: 1.7.4
Download
AMD64
Version: 1.7.4
Download
Linux
Package manager
Ubuntu/Debian
Copy
About Terraform
Featured docs
Introduction to Terraform
Configuration Language
Terraform CLI
Terraform Cloud
Provider Use
Terraform Cloud

app.terraform.io/app/example-org-a9a7ce/workspaces/getting-started
Workspaces
getting-started
Overview
Runs
States
Variables
Settings
example-org-a9a7ce
Latest Run
View all runs
Triggered via CLI
Applied
robin9899 triggered a run a minute ago via CLI
Policy checks
Add
Estimated cost change
Enable
Plan & apply duration
Less than a minute
Resources changed
+5 ~0 -0
See details
Resources 5
Outputs
Current as of the most recent state version.
Filter resources
NAME
PROVIDER
TYPE
MODULE
CREATED
prod_db
hashicorp/fakew...
fakewebservi...
root
Mar 8 2024
primary_lb
hashicorp/fakew...
fakewebservi...
root
Mar 8 2024
servers 0
hashicorp/fakew...
fakewebservi...
root
Mar 8 2024
Execution mode: Remote
Auto-apply API, CLI, & VCS runs: Off
Auto-apply run triggers: Off
Project: Default Project
Metrics (last 1 run)
Average plan duration < 1 min
Average apply duration < 1 min
Total failed runs 0
Policy check failures 0
Tags (0)
Add a tag
Tags have not been added to this workspace.