Chapter 6. Managing Secrets with Terraform

At some point, you and your software will be entrusted with a variety of secrets, such as database passwords, API keys, TLS certificates, SSH keys, GPG keys, and so on. This is all sensitive data that, if it were to get into the wrong hands, could do a lot of damage to your company and its customers. If you build software, it is your responsibility to keep those secrets secure.

For example, consider the following Terraform code for deploying a database:

```
resource "aws_db_instance" "example" {
  identifier_prefix   = "terraform-up-and-running"
  engine              = "mysql"
  allocated_storage   = 10
  instance_class      = "db.t2.micro"
  skip_final_snapshot = true
  db_name             = var.db_name

  # How to set these parameters securely?
  username = "???"
  password = "???"
}
```

This code requires you to set two secrets, the username and password, which are the credentials for the master user of the database. If the wrong person gets access to them, it could be catastrophic, as these credentials give you superuser access to that database and all the data within it. So, how do you keep these secrets secure?

This is part of the broader topic of *secrets management*, which is the focus of this chapter. This chapter will cover:

- Secret management basics