Step-by-Step: Install & Run Vault on Windows

✅ 1. Download Vault

Go to the official download page:

🔗 https://developer.hashicorp.com/vault/downloads

- Select **Windows (amd64)**
- Download and extract the ZIP (e.g., `vault_1.15.2_windows_amd64.zip`)
- Move `vault.exe` to a folder (e.g., `C:\Vault\`)

🔧 2. Add Vault to your PATH

1. Search "Environment Variables" in the Start Menu.
2. Click **"Edit the system environment variables"**
3. Click **Environment Variables**
4. In the "System Variables" section:
    - Find `Path`
    - Click Edit → Add `C:\Vault\` (or wherever you saved `vault.exe`)

Now you can run `vault` from any command prompt.

🚀 3. Run Vault in Dev Mode (Quick Start)

Open **Command Prompt** or **PowerShell**:

```
vault server -dev
```

You'll see:

- Vault running at: `http://127.0.0.1:8200`
- A root token — copy it

✅ 4. Set Environment Variables

Set these **each time** or use a script:

```
$env:VAULT_ADDR = "http://127.0.0.1:8200"
$env:VAULT_TOKEN = "paste-your-root-token-here"
```

Or in Command Prompt:

```
set VAULT_ADDR=http://127.0.0.1:8200
set VAULT_TOKEN=your-root-token
```

🔐 5. Store and Read Secrets

```
vault kv put secret/myapp username=admin password=1234

vault kv get secret/myapp
```

🏗️ Optional: Run Vault Securely (Standalone on Windows)

Create a config file `vault.hcl` (e.g., in `C:\Vault\vault.hcl`):

```
listener "tcp" {
  address     = "0.0.0.0:8200"
  tls_disable = 1  # only for dev/testing
}

storage "raft" {
  path    = "C:/Vault/data"
  node_id = "vault-node-1"
}

ui = true
```

Then run:

```
vault server -config="C:\Vault\vault.hcl"
```

In a **second PowerShell window**:

```
$env:VAULT_ADDR = "http://127.0.0.1:8200"
vault operator init
vault operator unseal  # run 3 times with 3 different keys
vault login <your-root-token>
```

✅ Use Vault with Terraform on Windows

1. Set env vars:

```
$env:VAULT_ADDR = "http://127.0.0.1:8200"
$env:VAULT_TOKEN = "your-root-token"
```

2. Create a `main.tf` file:

```
provider "vault" {}

data "vault_generic_secret" "example" {
```

```
  path = "secret/data/myapp"
}

output "secret_password" {
  value     = data.vault_generic_secret.example.data["password"]
  sensitive = true
}
```

3. Run Terraform

```
terraform init
terraform apply
```

📦 Step-by-Step: Store and Use Static AWS Keys from Vault

🔧 1. Store the AWS keys in Vault

From your terminal or PowerShell:`vault kv put secret/aws_creds access_key=AKIAxxxxxxxxxxx secret_key=xxxxxxxxxxxxxxx`

This writes the secret to path: `secret/aws_creds`

🧠 2. In Terraform, read that secret:

```
provider "vault" {
  address = "http://127.0.0.1:8200"
  token   = var.vault_token
```

```
}

data "vault_generic_secret" "aws" {
  path = "secret/data/aws_creds"
}

provider "aws" {
  region     = "us-east-1"
  access_key = data.vault_generic_secret.aws.data["access_key"]
  secret_key = data.vault_generic_secret.aws.data["secret_key"]
}
```

✅ 3. Run Terraform

Make sure you export your Vault token or set it in variables:

```
$env:VAULT_ADDR = "http://127.0.0.1:8200"
$env:VAULT_TOKEN = "your-root-token"
```

Then:

```
terraform init
```

```
terraform apply
```